

JC530 U.S. PTO
09/516699

日本国特許庁
PATENT OFFICE
JAPANESE GOVERNMENT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日
Date of Application: 1999年 3月 2日

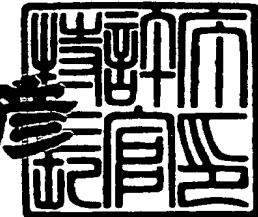
出願番号
Application Number: 平成11年特許願第053615号

出願人
Applicant(s): 株式会社日立製作所

1999年10月29日

特許庁長官
Commissioner,
Patent Office

近藤 隆彦



出証番号 出証特平11-3074588

【書類名】 特許願

【整理番号】 K98006721

【提出日】 平成11年 3月 2日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 19/00

【発明の名称】 ページ生成方法及び装置並びにページ生成プログラムを記録した記憶媒体および電子メールシステム

【請求項の数】 16

【発明者】

【住所又は居所】 神奈川県川崎市幸区鹿島田890番地 株式会社日立製作所 システム開発本部内

【氏名】 小池 博

【発明者】

【住所又は居所】 神奈川県川崎市幸区鹿島田890番地 株式会社日立製作所 システム開発本部内

【氏名】 岡山 将也

【発明者】

【住所又は居所】 神奈川県川崎市幸区鹿島田890番地 株式会社日立製作所 システム開発本部内

【氏名】 浪岡 美予子

【発明者】

【住所又は居所】 神奈川県川崎市幸区鹿島田890番地 株式会社日立製作所 システム開発本部内

【氏名】 増石 哲也

【発明者】

【住所又は居所】 神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所 ソフトウェア事業部内

【氏名】 林 重年

【発明者】

【住所又は居所】 神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所 ソフトウェア事業部内

【氏名】 嶋崎 康一

【発明者】

【住所又は居所】 神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所 ソフトウェア事業部内

【氏名】 吉田 祐一

【発明者】

【住所又は居所】 神奈川県川崎市幸区鹿島田890番地 株式会社日立製作所 システム開発本部内

【氏名】 井川 久美子

【特許出願人】

【識別番号】 000005108

【氏名又は名称】 株式会社日立製作所

【代理人】

【識別番号】 100068504

【弁理士】

【氏名又は名称】 小川 勝男

【手数料の表示】

【予納台帳番号】 013088

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 ページ生成方法及び装置並びにページ生成プログラムを記録した記憶媒体および電子モールシステム

【特許請求の範囲】

【請求項1】

ページ出力要求を受け付け、テンプレートとページ生成に必要なページ要素データから要求ページを生成して出力するページ生成方法であって、

所定のページ更新契機に従いページを生成しておき、該ページを出力することを特徴とするページ生成方法。

【請求項2】

請求項第1項のページ生成方法において、

始めのページは、データの更新契機により生成しておき、ユーザの要求を受け付けた際に該生成出力し、2回目以降の検索絞り込みページは生成することを特徴とするページ生成方法。

【請求項3】

請求項第1項のページ生成方法において、

最初のページを生成した際に、生成したページから次に生成する可能性のあるページを解析し、上記可能性のあるページを生成する次ページ解析生成制御機能を有し、次のページの送信要求がきた場合には、該可能性のあるページを出力することを特徴とするページ生成方法。

【請求項4】

請求項第3項のページ生成方法において、

次ページ解析生成制御機能は、入力となる最初に生成されたページ内部の個人に関する情報を認識する機能を持ち、認識した個人の情報を使用して、該個人にカスタマイズされた次ぎに可能性のあるページをバッチ生成することを特徴とするページ生成方法。

【請求項5】

請求項1から請求項4のページ生成方法において、

上記ページ要素データの更新契機と生成するページとの関係を定義したバッチ

ページ生成定義を入力とし、生成するタイミングと生成するページを決定することを特徴とするページ生成方法。

【請求項6】

ページ要求を受け付け、ページ生成に必要なページ要素データから要求ページを生成して出力するページ生成方法において、

上記ページ要素データの更新契機にページを生成し、ページ要求を受け付けた時に、該生成ページを出力することを特徴とするページ生成方法。

【請求項7】

ページ生成起動命令によりページ生成に必要なページ要素データから要求ページを生成するページ生成方法において、

該ページ生成起動命令を受け付け、要求されたページを、該バッチ生成ページの記憶から選別して、選別した該バッチ生成ページの内容を返すページ生成ステップと、

更新契機に従い該ページ生成ステップを該ページ生成起動命令により起動し、該ページ要素データからバッチ生成ページを生成し、該ページ生成手段が参照可能な位置に該バッチ生成ページを記憶するバッチページ生成制御ステップとを有することを特徴とするページ生成方法。

【請求項8】

ページ生成起動命令により指定されたテンプレートを読み込み、該テンプレート内の該ページ生成の命令を実行することで、ページ生成に必要なページ要素データを取得し、要求ページを生成するページ生成方法において、

上記ページ生成起動命令に、該バッチ生成ページであるテンプレートを指定しページ生成するページ生成ステップと、

該ページ要素データの更新契機に従い、該ページ生成手段を該ページ生成起動命令により起動し、該ページ要素データからバッチ生成ページを生成し、該ページ生成手段の読み込む該テンプレートとして該バッチ生成ページを記憶するバッチページ生成制御ステップとを有することを特徴とするページ生成方法。

【請求項9】

請求項6項から請求項8項のページ生成方法において、

上記ページ生成ステップでは、上記ページ生成起動命令を受け付けた時に、要求されたページの該バッチ生成ページが存在すれば、該バッチ生成ページを送り、存在しない場合は、該ページ要求のページを生成して送ることを特徴とするページ生成方法。

【請求項10】

ページ生成起動命令により指定されたページ生成の命令の集合であるテンプレートを読み込み、該テンプレート内の該ページ生成の命令を実行することで、ページ生成に必要なページ要素データを取得し、要求ページを生成するページ生成方法において、

上記ページ要素データの更新契機に従い、該ページ生成起動命令により起動し、第1のテンプレートを読み込み、遅延命令がないページ生成命令を実行するとともに、遅延命令がある命令から遅延命令を削除したページ生成命令を生成することで、第2のテンプレートを生成し、その中に記述されている該遅延命令を削除したページ生成命令を実行することで、要求ページを生成することを特徴とするページ生成方法。

【請求項11】

請求項1から請求項8のページ生成方法において、該ページ要素データの更新契機と生成するページとの関係を定義したバッチページ生成定義を入力とし、生成するタイミングと生成するページを決定することを特徴とするページ生成方法。

【請求項12】

請求項1から請求項8のページ生成方法において、上記更新契機として、生成されるページ内に使用するデータを記憶したデータベースの更新時、テンプレートの更新時、ページ生成するサーバの状態を変更するセットアップ時、指定時刻、指定時間経過時およびユーザ情報への更新時の少なくともいずれか一つを用いることを特徴とするページ生成方法。

【請求項13】

ページ要求クライアントからのページ要求を受け付け、要求ページを該クライアントへ送信するページサーバと、該ページサーバからのページ生成起動命令に

より、ページ生成の命令の集合であるテンプレートとページ生成に必要なページ要素データから要求ページを生成するページ生成手段とからなるページ生成方法において、

ページが生成された場合に次ぎに生成される可能性のあるページを生成されたページ内部のリンク情報から解析する次ぎページ解析手段と、バッチページ生成制御手段を起動し、上記の次ぎに可能性のあるページをバッチ生成させる次ページバッチ生成制御機能を具備し、次ぎのページの本来の要求がきた場合に、該生成したページの中から要求のあったページを転送し、また、バッチ生成されたページに要求ページがなかった時は、生成し、上記のページ解析手段へつなげることを特徴とするページ生成方法。

【請求項14】

商店クライアントと電子モールサーバと顧客クライアントがネットワークで接続されている電子モールシステムにおいて、

上記電子モールサーバは、U P L O A D 手段、バッチページ生成制御手段、ページ生成手段からなり、

上記U P L O A D 手段では、該上記商品クライアントからの商品データの転送を受け付け、内部の商品データのデータベースを更新し、商品データベースの更新が終了した段階で、バッチページ生成手段に更新契機を通知し、

上記バッチページ生成制御手段では、更新契機通知を受け取り、バッチ生成すべきテンプレートをバッチページ生成定義情報を使用して決定し、ページ生成手段を制御し、該テンプレート内部命令を実行することでページをバッチ生成し、新たなテンプレートとして記憶し、

上記ページ生成手段では、上記顧客クライアントからのページ要求があった場合、該新たなテンプレートを転送することを特徴とする電子モールシステム。

【請求項15】

ページ出力要求を受け付け、テンプレートとページ生成に必要なページ要素データから要求ページを生成して出力するページ生成装置であって、

所定のページ更新契機に従いページを生成する手段と、該ページを受け取り出力する手段とを有することを特徴とするページ生成装置。

【請求項16】

ページ出力要求を受け付け、テンプレートとページ生成に必要なページ要素データから要求ページを生成して出力するページ生成プログラムを格納した計算機読み取り可能な記憶媒体であって、

上記ページ生成プログラムでは、所定のページ更新契機に従いページを生成し、該ページを出力するページ生成プログラムを格納したことを特徴とする記憶媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、WWWで使用されるページの生成方法及び装置並びにページ生成プログラムを記録した記憶媒体および電子モールシステムに関する。

【0002】

【従来の技術】

近年、世界的なインターネット網の整備、及び、WWW技術の進展により、HTMLやXMLなどに基づいたページをインターネットを通してクライアントPCにローディングして表示するWWWページアクセス方法が普及している。また、固定化されたページの高速なアクセス方法として、以下の公知例1から公知例4がある。これは、ページをなるべくクライアントの近くにもってくるキャッシュの技術が中心にある。すなわち、ページをクライアントまで転送するコストの削減を行う技術である。

【0003】

一方、WWWページアクセス者の要求も多様化し、アクセス者や求めたい情報の指定により、WWWサーバサイトにおいて、動的にかつ機械的にページを生成するWWWページ生成方法が開発され、エレクトリックコマースにおける、電子カタログ等に応用されてきている。ページ生成方法を使用したページアクセス方法では、ページの生成コストと、生成されたページをクライアントまで転送するコストが必要である。

【0004】

ここでいうページとは、クライアントの画面を構成する情報のことである。WWWシステムでは、URLで呼出されWWWブラウザで表示される情報の単位であり、HTML言語で記述されたテキストデータや、XML言語で記述されたテキストデータ、GIF形式やJPEG形式の画像データ、動画データ等がある。WWWシステム以外の例では、クライアントPC上で動作するソフトウェアの1画面を構成する情報のことである。

【0005】

従来のページアクセス方法、および、ページ生成方法の手法には、以下のものがある。従来のページアクセス方法、及び、その高速化方法を以下に示す。

【0006】

公知例1：(WWWサーバの基本的な機能)

HTMLページをエディタ等のオーサリングツールを使用して、発信者自身が作成し、WWWサーバシステムに配置することで、ユーザのページ要求時に、該配置したページをユーザサイトに転送する方法。このWWWシステムの紹介は、例えば、コンピュータソフトウェア誌のVol.15、No2(1998)、pp.48-61に、「WWWにおけるインタラクティブな先読みシステムの設計と実装」というタイトルで掲載されている。図20が、この公知例のページアクセス方法を示すシステム構成図である。このWWW技術により、世界各地のクライアントからWWWページをアクセスできる方式が実現されている。

【0007】

ここで説明したWWWのページアクセス方法では、要求ページは発信者が変更しない限り固定化されてたものを想定しており、ユーザ情報やユーザ要求によりカスタマイズしたページを参照する技術に関するものでない。

【0008】

公知例2：(プロキシ方式)

ユーザの要求時、WWWページをプロキシと呼ばれるキャッシュシステムでキャッシュし、他のユーザ要求が発生した場合、WWWサーバに更新の否かを問い合わせ、ページの更新がなければプロキシ内のキャッシュのページを使用するペ

ージアクセス方法。このWWWシステムのプロキシ方式の紹介は、公知例1と同様、コンピュータソフトウェア誌のVol.15、No2(1998)、pp.48-61に、「WWWにおけるインタラクティブな先読みシステムの設計と実装」というタイトルで掲載されている。図21が、この公知例のページアクセス方法を示すシステム構成図である。このプロキシにより、WWWサーバのページ転送回数が減り、アクセス回線の太い、クライアントに対し身近なプロキシサーバから、ページの複製を転送することで、ページ要求を高速に処理する。このプロキシは、WWWブラウザの機能としてクライアントサイト内部に存在する場合もある。

【0009】

このWWWのページアクセス方法でもまた、要求ページは発信者が変更しない限り固定化されてたものを想定しており、ユーザ情報やユーザ要求によりカスタマイズしたページを参照する技術に関するものでない。

【0010】

公知例3：(プッシュ方式)

プッシュ方式としては、マイクロソフト社の Active Channel (アクティブチャネル) 技術がある。これは、プッシュサーバにCDF (Channel Definition Format) 形式のプッシュ定義を与え、これをプッシュクライアントが受け取り、クライアントの時間トリガ等により、ユーザページ要求とは非同期に、プッシュサーバのページコンテンツをクライアントのローカルキャッシュに読み込み、ユーザは、クライアントに送り付けられたページをクライアントのローカルキャッシュにあるページを参照することでアクセスする方法である。この技術は、富士通XML推進チーム執筆、「はじめてのXML」、日経BP発行、1997の第3章で示されている。図23が、この公知例のページアクセス方法を示すシステム構成図である。この方式は、ユーザがサーバへ接続しページを参照することなく、ページ自体が、各ユーザの知らない間に送り付けられれているため（実際は、クライアントが起動されているときに、時間トリガで、ページサーバへページ要求し、ページをユーザが意識しない間に転送処理をする機能である）、参照時はユーザのクライアント上にあるページを参照するため、高速にページを参照できる利点がある。このプッシュは、CDFの定義に従い時間トリガにより配布対象ページや配

布時間を指定できる機能をもつ。

【0011】

このページアクセス方法は、ページ参照が高速である反面、クライアントの蓄積物に対してインタラクティブに情報の問い合わせができない点や、また、プッシュサーバは、ページのバリエーションも多数のクライアントへ送り付けるため、大勢の人向けのページを準備して選別して配布するしかできない。また、配布するCDF定義の内容をユーザごとにカスタマイズし、クライアントに配布して、異なるページの参照を行えるが、これにおいても、ページ自身は、固定的に作成されたものをプッシュすることにはかわらない、ユーザ情報やユーザ要求によりカスタマイズしたページ（例、検索処理ページ）できないという課題がある。

【0012】

公知例4：（ページ先読み方式）

また、利用者からみたページアクセスの応答時間を短縮する他の例が、コンピュータソフトウェア誌のVol.15、No2(1998)、pp.48-61に、「WWWにおけるインタラクティブな先読みシステムの設計と実装」というタイトルで掲載されている。これには、WWWページの先読みシステムの例であり、先読みシステムには、（1）連想型と、（2）統計型がある。（1）の連想型は、ユーザが要求したページを転送後、そのページ内部をスキャンして、次に要求される可能性のあるページを類推し、ユーザ操作とは独立にプレ呼出しを実行し、その結果えられた先ページをキャッシュに記憶することで、ユーザの操作で先ページが要求された場合に、キャッシュから取り出し高速にユーザに表示する技術である。図24がこのページアクセス方法を示す図である。この技術は、ネットワーク的に遠いサイト（ロードに時間がかかるサイト）の情報を、ネットワーク的に近いサイト（要求時高速に取り出せるサイト）に、ユーザが要求する前に取得しキャッシュ化しておくことを目的とした技術であり、プロキシサーバや、ユーザクライアントでの実装について論述されている。また、このプレ呼出しの結果得られたデータは、そのプロキシを使用するユーザで共有されることを目的に、キャッシュ中で共有され、他の人が同じページを参照した場合にも、高速にページを参照できるしくみが紹介されている。各ユーザが、別々に要求するユーザごとに異なるペー

ジに対しては、この機能の適用は記述されておらず、適用外である。また、(2)の統計型は、バッチ型先読みとして紹介されており、プロキシで処理したページ情報を統計処理して、アクセスがあるだろうページを類推し、定期的なバッチページ呼出しの実行により、それらのページを事前にプロキシに取り込む技術である。しかし、統計的に参照される可能性のあるページの事前取得であり、トラフィックを増大させるわりに、キャッシュの効果がきたいできず、WWWサーバばで、再度ページ要求することがおおい。

【0013】

上記の公知例1、2、3、4は、いずれもサーバで固定化されサーバに配置されたWWWページの参照方式や、固定ページの参照の高速化に関する公知例であった。次にユーザからの異なる要求ごとに、ページを動的に生成するページ生成方法の公知例を以下に示す。

【0014】

公知例5：(動的なページ生成方式)

ユーザ情報やユーザ要求によりカスタマイズしたページを作成するWWWサーバ技術は、CGI(コモンゲートウェイインターフェース)としてしられている。これは、ユーザ要求をうけて、サーバで処理し、HTMLページを合成してユーザに転送する技術である。このページ生成を一般化したものがページ生成手段である。ページ生成手段は、WWWページのユーザの要求毎に指定されたテンプレート(HTMLの雛形と処理スクリプト)を解釈し、DBのデータを取得し、その値をテンプレート内に、埋め込むことで動的にHTMLページを生成する。生成されたページは、WWWサーバ経由でユーザのクライアント上のWWWブラウザに転送され表示される。ページの動的な生成に関する公知例として、仮想的な商店街において商品検索方法に関する発明「物品検索方法及び装置」(特開平8-22498)や、ユーザ毎にカスタマイズしたWWWページを動的に作成する方法に関する発明「情報を検索するコンピュータによる実行方法及び装置」(特開平9-26970)がある。図2.2が、本公知例のページ生成方法を示すシステム構成図である。本公知例では、データベースと連動し、発信者が、生成されるページの雛形であるテンプレート記述物を作成するだけで、ユーザが要求するごとに、D

B 内部の多量の商品データを元にした商品ページを動的に生成し、ユーザに転送できる利点がある。1商品に対し1つのページを手作りに作成していた、公知例1、2、3、4に比べ、ページ生成の自動化が行え、発信作業にかかるページ作成コストを大幅に削減できる。また、ユーザの問い合わせ情報から固定的には作成できないページ（例、ユーザの名前を入れる等）を生成できる利点もある。但し、動的なページ生成方式は、ページ要求があるごとに毎回、必要なデータの取得やページテンプレートの解析、そして生成と、ページ生成に時間がかかるという課題がある。

【0015】

上記、公知例1、公知例2、公知例3、公知例4、共に、高速なページアクセス方法を実現する技術を提供しているが、ユーザの要求に対しフレキシブルなページを提供する、動的なページ生成方法における高速なページ生成方法、及び、動的に生成されたページの高速なページアクセス方法について述べたれていない。また逆に、公知例5は、動的なページ生成についての技術であるが、その高速化に関して検討されていない。現在、動的に生成されるページを高速にアクセスするためのページ生成方法が要望されている。

【0016】

【発明が解決しようとする課題】

従来のシステムの、公知例1、公知例2、公知例3、公知例4では、固定化されたページを高速にアクセスする技術が記述されているが、反面、転送するページは、事前に作成され固定化されたものが対象であり、ユーザがページをアクセスする時点で、動的に作成する必要のあるページを、高速にアクセスする技術や、ページ生成自体の高速化について記述されていない。

【0017】

一方、公知例5のような動的なページ生成方式の公知例では、テンプレートと埋め込むデータから各種ページを動的に生成するため、フレキシブルなページ生成ができる。しかし、毎回、テンプレートの解析、データのデータベースからの検索&取得、ページの合成を行うため、ページを取得するまでのページアクセス時間が遅いという課題があった。さらに、この動的ページ生成方式では、公知例

2のプロキシが有効に働くか、毎回、ユーザの要求の全てが、サーバでの動的なページ生成となるという課題があった。また、従来の公知例5の動的なページ生成では、ある時間範囲をみると、結果まったく同じページが生成される場合でも、要求単位にサーバで各種処理を行いページを生成するため、ページ生成要求を同時に多数の異なるユーザから受け付けると、ページ生成に時間とサーバのリソースを奪われ、さらにページ生成が遅くなるという課題がある。

【0018】

本発明の目的は、フレキシブルなページを生成するため、動的なページ生成を行い、ユーザの要求に対応し生成されたページを転送するページアクセス方法における、高速なページ生成を実現するページ生成方法を提供する点にある。

【0019】

【課題を解決するための手段】

本発明では、ページ出力要求を受け付け、ページ生成の命令の集合であるテンプレートとページ生成に必要なページ要素データから要求ページを生成して出力するページ生成方式を用い、所定のページ更新契機に従いページを生成し、該ページを出力する。

【0020】

また、本発明では、ページ生成起動命令によりページ生成に必要なページ要素データから動的に要求ページを生成するページ生成方式を用い、

該ページ生成起動命令を受け付け、要求されたページを、該バッチ生成ページの記憶から選別して、選別した該バッチ生成ページの内容を返すページ生成ステップと、

該ページ要素データの更新契機に従い、該ページ生成手段を該ページ生成起動命令により起動し、該ページ要素データからバッチ生成ページを生成し、該ページ生成手段が参照可能な位置に該バッチ生成ページを記憶するバッチページ生成制御ステップとを用いることにより上記課題を解決できる。

【0021】

上記の解決手段により、動的に変化するページの本来の変化のタイミングで固定ページ化することができ、固定のページに対する上記公知例の高速化ページア

クセス方法が適用できるとともに、毎回、動的に生成されるページと同じ内容のページをユーザがアクセスできる方法が提供できる。

【0022】

公知例1、2、3、4と、本発明は、補完関係にあり、特にバッチ生成されサーバに記憶されたページは、公知例1、2、3、4とともに使用することで、ページをクライアントまで転送するコストが削減され、さらなる高速でかつフレキシブルなページアクセスの方法を提供できる。

【0023】

【発明の実施の形態】

本発明は、WWWで使用されるHTMLページのうち動的にかつ機械的に生成されるページのページ生成方法に関わり、ページの内容のライフ時間に合った効果的なページ生成を行う方法に関する。すなわち、特に、長いライフ時間でページが変化せず、1ページの生成に時間のかかるページのアクセスを高速化する方法に関する。

【0024】

以下、本発明の一実施例のページ生成方法について、図1から図11を参照しながら説明する。

【0025】

図1は、本発明のページ生成方法を、WWWページアクセスシステムへ適用した場合の一実施例であり、制御とデータの流れを示したWWWページアクセスシステム構成図である。図2は、図1の本発明のページ生成方法を適用したWWWページアクセスシステムが動作する装置のシステム構成である。図3は、図1のシステムの動作を示すフローチャートである。図4は、ページ生成の雛形であるページ生成手段へ入力するテンプレートの記述例である、図5は、図4のテンプレートから生成されたHTMLページの一部の例、図6は、図5のHTMLのWWWブラウザにおける表示例である。また、他のページの例として、ページに埋め込むデータの例を図7のデータテーブルで示し、図8、図9に、図7のレコードデータを使用した場合の生成されたHTMLのWWWブラウザでの表示例、図10は、図8の画面に対するページ生成の事例であり、テンプレートとデー

タから生成されるH T M Lの関係を示した図である。図11は、ページ生成をバッチ制御するバッチ生成定義の例である。

【0026】

まず、本実施例であるWWWページアクセスシステムが動作する装置のシステム構成を図2を用いて説明する。1は本発明の一実施例のWWWページサーバ処理を行うサーバコンピュータであり、3は、ページ要求するユーザのパーソナルコンピュータであり、2の、インターネットやイントラネットのネットワークで、サーバコンピュータ1と各パーソナルコンピュータ3は接続されている。パーソナルコンピュータ3は、ユーザからのページ要求の受付やサーバコンピュータ1へのページ生成依頼、生成されたページを表示することを目的に、処理を担当するコンピュータとページを表示するディスプレイと入力デバイスであるキーボードやマウスポインタ、ネットワークアダプタ38で構成されている。また、コンピュータ内には、OS等のソフトウェアとともに、WWWページを表示したり、要求したりするためソフトウェアとしてWWWブラウザ35が動作する。2のネットワークは、ユーザのページ要求をパーソナルコンピュータ3からページ生成サーバであるサーバコンピュータ1へ伝えたり、生成したページをパーソナルコンピュータ3に送ったりするためのもので、WWWの場合、H T T Pプロトコルで通信データが交換される。通信データの交換は、サーバコンピュータ1の通信手段36とパーソナルコンピュータ3の通信手段37により、ネットワークアダプタ8や38、ネットワーク2を用いて行われる。本発明のページ生成方法を実現するサーバコンピュータが1である。1は、ページ生成処理等を行うC P U 5、データを保存するためのディスク記憶装置7、パーソナルコンピュータ3とネットワーク2をかいして接続するためのネットワークアダプタ8、C P U 5により実行されるページ生成のための各種手段（プログラム）と、各手段により使用される各データを記憶するメモリ4、サーバコンピュータの状態等を表示、指示するためのディスプレイと入力デバイス9からなり、それら、4、5、7、8、9がバス6により接続され、ページ生成のためのサーバコンピュータ1を構成している。7のディスク記憶装置内には、TEMP LATE群30、データ22の集まりであるデータテーブル31、バッチページ生成

定義データ20の集まりであるバッチページ生成定義32、及び、生成されたpage24の集まりであるPage群33が記憶される。4のメモリ上には、本発明のページ生成方法を実現する手段として、OS手段21、WWWサーバ手段11、ページ生成手段12、バッチページ制御手段13、データ更新手段14、DBMS（データベースマネージメントシステム）手段15、通信手段35が記憶され、CPU5により実行される。OS手段21は、サーバコンピュータのオペレーティングシステムである。WWWサーバ手段11は、パーソナルコンピュータ3のWWWブラウザ35からの要求を受けて、Page群33の中の指定のPage24をWWWブラウザ35に送ったり、動的なページ生成が必要な場合には、ページ生成手段12にページ生成の命令を行う処理を行う。ページ生成手段12は、WWWサーバ手段11やバッチページ生成制御手段13からのページ生成命令により動作し、外ページ生成命令で指定されたTEMPLATE23をTEMPLATE群30から参照し、解釈実行し、解釈中ででてきたデータ要求命令に従い、DBMS手段にデータ22の参照を要求し、得られたデータ22とTEMPLATE23の雛形部分へデータを展開することで、Page24をページ生成し、要求もとの11や13の手段に返す。バッチページ生成制御手段13は、バッチページ生成定義32の中のバッチページ生成定義データ20を入力として、タイマ手段16の時間トリガや、DBMS手段15が管理するデータテーブル31のデータ22を更新するデータ更新手段14が更新するとに発行する更新トリガを受付、バッチページ生成定義に従い、ページの生成をページ生成手段12に命令する。本システムにより、ユーザ要求毎に毎回ページ生成されていたページの変更サイクルに着目し、ページを事前生成し固定化しておくことで、ユーザ要求があった場合、毎回生成しているページと同じページを、固定ページとして提供することで高速なページ生成と、ネットワーク上のプロキシを有効にした高速なページアクセス方法を提供する。

【0027】

以下、図2のシステム構成で示した本発明のページ生成方法によるWWWページアクセスシステムの動作事例を、図1の制御フローとデータフローが示されているシステム構成図、および、図3のフローチャートで、詳細に説明する。

【0028】

図3のフローチャートにおいて、本実施例の処理の開始は2個所存在する。まず、動的に生成されるページが、ページを事前にバッチ生成するフローチャートである（ステップ1000から開始）。もう一方は、ユーザがページ要求をしてページを表示するまでのフローである（ステップ1020から開始）。

【0029】

まず、ステップ1000は、ページの構成に関わるデータが更新される場合に開始される。データ22のデータ更新を行うデータ更新手段14がデータ22をDBMS手段15に命令し更新する（ステップ1001）。このデータ更新手段14は、例えば、商品データを管理するソフトウェアであったり、データベースシステムのデータ保守ツールであったり、テンプレートを更新するソフトウェアであったり、インストーラやセットアップソフトウェア等である。このデータ更新手段14が、データ22のあるデータを更新したタイミングで、データ更新手段14は、データの更新があったことをしめす更新トリガを、バッチページ生成制御手段13へ通知する（ステップ1002）。更新トリガの通知手段は、バッチページ生成制御手段13を、トリガの種類を引数に起動する方法や、バッチページ生成制御手段13のメソッドをコールする方法がある。次に、通知を受け付けたバッチページ生成制御手段13は、バッチページ生成定義データ20から生成すべきページを決定する（ステップ1003）。次に、この決定でえられた、テンプレートとデータ識別子を指定して、ページ生成手段23でページを生成する（ステップ1004）。次に、生成したページ24がWWWサーバ手段11から参照できるサーバコンピュータ1のディスクアクセス記憶装置7に記憶するファイル名称を、バッチページ生成定義データ20の生成ページ名から決定する（ステップ1005）。次に、生成したページを決定したURLで参照される位置の、ディスク記憶装置7上の位置に記憶する（トップ1006）。これで、ステップ1000から始まるページのバッチ生成処理は終了する（ステップ1007）。

【0030】

次に、図3のもう一方の、パーソナルコンピュータ3で、ユーザがページ要求

をしてページを表示するまでのフロー（ステップ1020から開始）を、詳細に説明する。

【0031】

ユーザは、パーソナルコンピュータ3の入力装置（キーボードやマウス等）を使用して、WWWブラウザ35を操作してページを要求する（ステップ1021）。次に要求ページが、要求ごとに変わるページか、変化がすくないページか（商品データなどは、データの入れ替えがない場合同じページとなる）を判定する（ステップ1022）。次に、固定のページへのURLの場合、変化が少ないと判定し（ステップ1022）、パーソナルコンピュータ3のWWWブラウザ35は、生成済みページの要求（例、http://www.aaa.co.jp/a_g_I000a.html）をネットワーク2経由で出す（ステップ1023）。ステップ1023による要求をページサーバ処理を行うサーバコンピュータ1内のWWWサーバ手段11が受け取り、前記ステップ1006で記憶されたバッチ生成されている、ページ要求指定されたページ24をWWWサーバ手段11がネットワーク2を通してパーソナルコンピュータ3のWWWブラウザ35に送信する（ステップ1024）。この場合、図1のプロキシ40も効果的にはたらき、バッチ生成されたページ24と同じページがプロキシ40のキャッシュ41にある場合は、プロキシ40に記憶されているページを使用でき、更新チェックをプロキシ40とWWWサーバ手段11の間で行うだけで、サーバコンピュータ1からページを転送しなくとも高速にページを転送できる。転送されたページは、パーソナルコンピュータ3のWWWブラウザ35により処理され、ディスプレイに表示され（ステップ1025）、ページ表示が終了する（ステップ1026）。この場合、ユーザ要求がかかる前に、ページを構成するデータ22やテンプレート23の変更をトリガとして、サーバで事前にバッチ生成されているので、実際にユーザがページをアクセスする場合に、毎回動的に生成するページと同じページが、固定ページとして参照でき、さらにプロキシと協力した高速なページアクセスが実現される。

【0032】

次に、図3のステップ1022における毎回変わる場合のフローを説明する。ページ生成手段23での生成要求をだす（ステップ1030の）。WWWサーバ

手段11はその要求をネットワーク2を通して受け付け、ページ生成手段23を起動する（ステップ1031）、次に、ページ生成手段23で、DBMS手段15と連動し、テンプレート23とデータ22から動的にページを生成し（ステップ1032）、動的に生成されたページ24をWWWサーバ手段11がネットワーク2を通してページ要求があったパーソナルコンピュータ3のWWWブラウザ35へおく（ステップ1033）。転送されたページ24は、WWWブラウザ35により処理され、ディスプレイに表示され（ステップ1025）、ページ表示が終了する（ステップ1026）。以下、上記ステップを以下で補足する。

【0033】

まず、ステップ1002における更新トリガについて説明する。更新トリガは、バッチページ生成のタイミングを通知するための情報である。タイミングと考慮されるものは、指定時間や、テンプレートの更新、ページを構成するデータの更新の情報であり、さらに、その時の属性（例えば、時間なら何時何分、データならどの商品データか等）を情報としてもつ。これらは、タイマ手段16や、データ更新手段14、また、インストーラやセットアッププログラム、また、サーバ管理者からのコマンド等のプログラムから、バッチページ生成制御手段13へ伝えられる。伝える方法は、OSへメッセージ交換ソフトウェアの機能を使用したり、共用メモリ機能を使用する。この更新トリガ機能の発生を、バッチページ生成手段は、どのページが変更すればいいかの判断をするトリガをもらい、かつ、その判断のための属性値を取得できる。これにより、バッチページ生成制御手段13は、生成するべきページを判断する機械と判断材料が得られ、ページ生成手段11を使用したページのバッチ生成が行える。

【0034】

次ぎに、ステップ1003におけるバッチページ生成定義データについて説明する。バッチページ生成定義データは、上記の更新トリガと生成すべきページ（実際は、生成するバッチページ生成命令）との関係を規定したデータである。バッチページ生成定義データ20の一実施例を図11の71、72、73等で示す。図11の各レコード71や72等は、各バッチページを行う条件（トリガの種類、トリガの成立条件）と、条件が成立した場合のバッチページ生成命令、生成

したページをWWWサーバが参照できる位置に記憶するときのHTMLファイル名で構成される。トリガの種類は、動的に生成されるページ24の構成要素（データ22やテンプレート自体23）が変更されるトリガの種類でありデータ更新手段14やのタイマ手段16から通知される。トリガの成立条件とは、バッチページ生成命令（バッチコマンド）が実行される条件が記述されている。起動バッチコマンドとは、ページ生成をバッチ起動するコマンドが記述されている。ステップ1003において、例えば、トリガがデータ更新で、成立条件がA商品データ更新が成立した場合に、生成すべきページとして、「page TEMPATE=a.html&ITEM=I000a」コマンドが決定される、すなわち、テンプレートがa.htmlで、その内容は、商品番号I000aのカタログページであることが決定される。バッチページ生成定義データ70の図11の例の71は、ページテンプレートの内部に時間に依存する処理やその値の定義がある場合の例であり、73は、テンプレート自身の変更のために、それに関連して生成されるページ全てのデータに対してバッチページ生成を行うことを示している（命令の中野、ITEM=*がそれを示し、*が条件をしめしている。*は、全てのデータに対して行えという意味である。*の代わりに、条件指定してもよい。例、ITEM="分類=アウトドア"。この例では、データの分類項目がアウトドアであるデータが選ばれ、全てバッチ生成される）上記のごとく、バッチ生成定義データ20により、バッチページ生成制御手段13は、更新トリガとバッチ生成ページとの関係、及び、生成したページの記憶場所の情報により、ページ内容のライフサイクルに合わせたページ生成を行える。

【0035】

次に、ステップ1004におけるバッチページ生成を詳細に説明する。本事例では、商品ページの検索を例に、ページ生成をバッチで起動制御してページを生成する過程を説明する。本実施例では、バッチページ生成制御手段は、「page TEMPATE=a.html&ITEM=I000a」コマンドを実行することでページ生成を起動する。図5がバッチ生成後のページの一部であるHTMLである。図6が図4により生成されたページ（図5）の実際のWWWブラウザ上での表示例である。図4に示すテンプレートは、サーバコンピュータ1のディスク記憶装置7に記憶されて

おり、1つのテンプレートあたり1つのファイルになっている。本実施例で述べているテンプレートは、HTMLページを生成するため、HTMLのページ記述中にページデータ生成命令が埋め込んだものである。図4の40001から40020がページデータ生成命令である。40002等のECEX文は、データテーブル31のRDBを検索する命令であり、結果あられた値が\$KEYWORD、\$NCONDに代入され、40004、40005の文でHTML文中で展開される。40003の文と40007の文のREPEAT文は、検索結果が複数のレコードの場合に、そのレコードの数だけループして、生成を繰り返す。生成されたページの結果の一部が図5である。40030の文や40031の文が変数値が展開されたHTMLであり、その他の文は、テンプレートにあったHTMLが使用されている。また、WWWブラウザで表示すると図6のように表示される。上記のごとく、バッチページ生成はおこなわれ、テンプレートとデータから目的のページがバッチ生成される。

【0036】

次ぎに、ステップ1005におけるバッチ生成したページを記憶するする場所（ファイル名やパス名）を決定する方法について述べる。前記のバッチページ生成定義データを例にすると、生成ページ名「a_g_I000a.html」により、生成したページのファイル名をa_g_I000a.htmlに決定する。この決定方法は、テンプレート名所とアイテム名所の組み合わせで名づける命名規則を使用している。次に、生成したページを決定したURLで参照される位置の、ディスク記憶装置7上の位置に記憶する（トップ1006）。今回の例では、URLとして参照できる「http://www.aaa.co.jp/a_g_I000a.html」に配置する。このように、バッチページ生成命令の要素から生成ページ名を構成できる命名規則を採用すると、自動的に、生成ページ名所を生成できる利点がある。また、図11の例における、生成ページ名の欄にファイル名所を記憶しておく方法も考えられる。この方法によれば、定義時に指定しておくことで、前記の名前の合成処理を軽減できる効果がある。

【0037】

次ぎに、ステップ1022における毎回変わるページか否かの判定方法を述べ

る。本実施例では、この判定を要求時に、固定のページへのURL（例、http://www.aaa.co.jp/a_g_I000a.html）の場合と、動的に生成する場合のURL（<http://www.aaa.co.jp/cgi-bin/page? TEMPLATE=a.html&ITEM=I000a>）の記述の違いで判定する。このリンク情報の違いは、ページサーバのページ生成機能が、1つ前のページ生成段階で、バッチ版か、動的生成版のいづれかを決め（実際はテンプレートの記述でページの発信者が事前に決定する）、ページのリンク情報として生成し、ページをクライアントに転送することで、使い分けさせる方法がある。これは、発信者が、テンプレートの記述を変えることで行える。上記のように、従来すべて動的に生成していた検索等のページを、ユーザのインタラクションに依存しなくページサーバのデータやテンプレートの更新のみに依存するページと、毎回異なるページの、2つの集合に分類し、前記のページ集合は、サーバでバッチ生成させ、後者はよびだされる都度生成するように、サーバコンピュータ1のテンプレート群30のWWWページテンプレート間のリンク関係を設定する。これにより、発信者は、毎回、ユーザかの呼出し毎に生成していたページを、ページ生成の時間のかかり方と、ディスク装置のメモリ量とのトレードオフにより、バッチページ生成する対象を指定でき効果がある。

【0038】

次ぎに、ステップ1030。例として図4の検索ページを例にする。図4のテンプレートを使用し、ページ生成手段23でページ生成するときのコマンドの引数として検索条件を指定しないでページを作る。これにより、図5、および図6のようなHTMLが作成される。この図4のテンプレートは、検索条件に従い、ページのデータテールル31を検索し、その現在存在するレコードの値を求め、その項目値を集合要素として取り出すことを行う。すなわち、図6の例では、データテーブルから取り出されたデータ22のレコードは、184件あり、その大分類50といわれる項目の値が、「文書、音声、画像、動画」のいづれか1つを持っている。条件指定（\$CONDITION）が空で検索のページの初期ページが作成される。この検索の初期ページ（図6）は、ユーザすべてに同じであり、商品データの更新に依存して変化する。図4の検索のため初めて使用されるページは、商品データの更新によりバッチ生成して、記憶することができる（ステップ1

000からステップ1007)。しかし、実際の検索時は、ユーザが図6の選択候補をマウスで指定することで、検索条件が指定され、初期ページをバッチページ生成した上記テンプレート同じテンプレートとを、動的に呼び出し、検索条件が加わり検索の絞り込みを行う。これにより、素ぼり込まれた候補が、又、図6のような表となる(但し、絞り込まれた分だけ、選択候補が現象する)。上記のように、検索の候補値を絞り込む場合、その初期値は、ユーザ毎に変化がなく、サーバ内部のデータの更新のみに依存して変化する。この初期ページは、一般に候補値を得るために検索時間を多く必要とする。これらを考えると、バッチページ生成方法は、この検索初期値ページ(候補値を検索するデータ内部から得る方法に基づく)に対して、有用に動作する。

【0039】

最後に、ページ生成手段の動作を他の事例を使用した、図7、図8、図9、図10を使用して詳細にその動作を説明する。図10のテンプレート65を、引数「ITEM=I000a」バッチ生成する。バッチ生成手段は、図10の65のテンプレートを読み込み、ページ生成しながら、アンダーラインの命令を実行する。まず、EXEC命令を実行し、図7のテーブルのデータを ITEM=I000aで検索する。結果、

「商品名=“クリスマスディナー券”」等のデータが得られる。これらの得られた値が「\$商品名」等の変数の位置で展開される。図10において、ITEM=I000aに対するデータ66(実際は、図7の61のレコードデータ)が決定され、ページ生成手段は、\$XXXの変数名の部分にデータを展開する。展開の結果生成されるページ67とテンプレートの命令との間の関係を図10は示している。WWWブラウザで表示した例が図8である。また、「ITEM=I000b」の場合(実際は、図7の62のレコードデータ)の例では、図9のような生成ページとなる。

【0040】

本発明のページ生成方法を採用したWWWサーバシステムは、上記のごとく動作するため、フレキシブルのあるページをつくるために動的にページを生成する方法においても、サーバで生成されるページを事前に生成することができ、実際のユーザがページを要求する時に動的に生成するより高速にページをアクセスできる効果がある。

【0041】

上記実施例は、バッチ生成したページをWWWサーバから直接参照できる位置に配置してクライアントからアクセスする方法する実施例であったが、本発明は、これに限定されるものでない。バッチ生成されたページも、動的に生成されるページと同じように、ページ生成を再度実行してクライアントへ転送してもよい。すなわち、バッチ生成されたページ 자체を、ページ生成手段のテンプレートとして扱う方法である。この場合の利点は、第1のバッチ生成で、テンプレートのある部分が固定ページにおきかえられ、第2のページ生成じに、実際のページ生成の処理量を削減できる効果がある。下記の命令は、ページ生成型のページ要求のURL記述である。

【0042】

`http://www.aaa.co.jp/cgi-bin/page? TEMPLATE=a.html&ITEM=1000a`

上記の命令をバッチページ生成した後に格納したファイルなa_g.htmlとすると、

`http://www.aaa.co.jp/cgi-bin/page? TEMPLATE=a_g.html11&ITEM=1000a`

で、同じページ内容でかつ高速にアクセスできる。

【0043】

この方法の利点は、すべてのページが動的に生成されるようになっている場合で、ページの生成が極端に時間がかかるような時、簡単に、リンク関係を「_g」を追加して、バッチページ生成を起動するだけで、ページ生成方法が使用でき、高速化されたページ参照が実現できる利点がある。また、他の利点としては、a.htmlテンプレートからバッチ生成される a_g.htmlがなければ、a.htmlを使用して動的にページ生成して転送するようにもできる。すなわち、該ページサーバがページ要求を受け付けた時に、要求された該バッチ生成ページが存在すれば、該バッチ生成ページをクライアントに送信し、存在しない場合は、該ページ要求のページを動的に生成してクライアントに送信するページ生成の切り替え手段をもつことも可能である。この場合、バッチ生成が起動中で生成後のページ格納ファイルがロックされている場合、ページ生成から参照できなくなるが、この場合の代替えとして本来のページ生成を行うことで代行を行える効果がある。

【0044】

上記実施例は、テンプレート23の内部の命令が一回のページ生成ですべて実行されHTMLに展開される実施例であったが、本発明は、これに限定されるものでない。すなわち、図12のように、第1のテンプレートA101の内部にページ生成を遅延させる命令をもたせ、ページ生成手段に、この命令遅延命令を処理する機能をもたせる。この遅延命令付きページ生成手段100は、該第1のテンプレートAをバッチページ生成し、第2のテンプレートA'102を作成し、実際のユーザからのページ要求時に、第2のテンプレートA'102の内部の遅延命令が取れた命令を実行し、ユーザ要求固有ページを動的に生成し転送する方法である。本実施例では、テンプレートの記述命令に、以下のような実行遅延命令をもたせる。

【0045】

例1： <!@non ECEC INTERFACE=PDBManager Method=GetRead Param=(a>1) >

例2： <!@non \$商品名>

この「non」命令を含む第1のテンプレートA 101が、バッチページ生成制御27からの呼出しで遅延命令付きページ生成手段100により、バッチページ生成されると、遅延命令が外された。

例1： <!@ ECEC INTERFACE=PDBManager Method=GetRead Param=(a>1)>

例2： \$商品名

になり、第2のテンプレートA' 102の内部で実行される命令になる。実際のユーザ要求ページが来たとき、遅延命令付きページ生成手段100が、上記命令を含むテンプレートA'を実行し、ページを動的に作成する。上記を行うために、遅延命令付きページ生成手段100は、命令を遅延させる命令を解釈できる機能を具備している。また、遅延命令による実現の他の実施例では、第1のテンプレートA 101のページ生成のさい、第2のテンプレートA' 102を作成する段階で、第2のテンプレート内部に展開する命令文を生成する方法がある。以下のその際の文字列合成命令の例である。

【0046】

<!@sprintf("<!@ ECEC INTERFACE=PDBManager

Method=GetRead Param=(a>1)">

この実施例では、積極的にページ生成命令を合成するため、サーバの状態に応じた第2のテンプレートA'の合成が可能となる。

【0047】

上記実施例を実現するサーバのシステム構成を図13に記述する。図2との違いのみ説明する。まず、ページ生成手段に遅延命令を処理する機能が具備されている。この遅延命令付きページ生成手段100は、TEMPLATE群30からメモリ4中に読み込まれたTEMPLATE A'を解釈実行し、第2のテンプレートA'を、メモリ4に生成する。この生成したテンプレートをさらに、バッチページ生成手段13が、ディスク記憶装置7のTEMPLATE群30の記憶エリアに記憶する。これにより、TEMPLATE A' 102も、遅延命令付きページ生成手段100のテンプレートとして使用できるようになる。実際に、パーソナルコンピュータ3からユーザのページ要求がきた場合は、TEMPLATE A' 102を使用し、動的にページを生成して、パーソナルコンピュータ3へ、転送する。この動的な生成において、サーバの状態にみに依存する部分は、ユーザがページ要求する前にバッチ生成されているため、実際の動的なページ生成では、ユーザの固有情報に基づくもののみを動的に生成しページを組み立てるため、ユーザ情報に基づくページ生成であっても、ページ生成が高速になる効果がある。

【0048】

さらに、図14のフローチャートを使用して、遅延命令付きバッチページ生成の動作を詳細に説明する。第1の実施例と同様、サーバコンピュータ1のデータ22の更新やタイマートリガ等のサーバ状態の変化により開始される（ステップ2000）。まず、データ更新手段15がDBMS手段15が管理しているデータテーブル31のデータ22を更新する（ステップ2001）。次に、データ更新手段14は、更新トリガをバッチページ生成制御手段13へ通知する（ステップ2002）。通知を受けたバッチページ生成制御手段13は、バッチページ生成定義データを参照し、生成すべきページの生成方法（バッチコマンド）を決定する（ステップ2003）。決定されたページ（ページ生成コマンド）を、テ

ンプレートAとデータを指定してページ生成手段100で生成する。その際、遅延命令があるテンプレート内の命令は、遅延命令が実行され遅延命令が削除された通常の命令文としてHTMLのテンプレート内部に生成する（ステップ2004）。バッチページ生成制御手段13は、生成ページがテンプレートA‘として記憶する名称と位置を、バッチページ生成定義データ20から決定する（ステップ2005）。最後に、生成したページ（実際は、テンプレート）をページ生成制御手段13が、ページ生成手段100が参照するテンプレートとしてディスク記憶装置7の中のTEMPLATE群30のエリアに記憶する（ステップ2006）。以上で、データ更新により開始された、ステップ2000から2006を終了する（ステップ2007）。次ぎに、実際にユーザがページを要求する場合の処理フローを、ステップ2020からステップ2037で説明する。まず、ユーザ、パーソナルコンピュータ3のWWWブラウザ35からページ要求を行う（ステップ2021）。ユーザのページ要求は、どのマシーンのどのテンプレートを使用したどのデータにより生成したページかを指定する。この要求は、URLの指定で行う。例えば、「<http://www.aaa.co.jp/cgi-bin/page?TEMPATE=ad.html&ITEM=I000a>」である。

【0049】

この命令を、<http://www.aaa.co.jp/>で識別されるマシーンのWWWサーバ手段11へ、ネットワーク2経由で要求を出す（ステップ2022）。次ぎに、WWWサーバ手段11は、要求を受け付けると、ページ生成手段11を起動する（ステップ2023）。ページ生成手段100は、テンプレートA'「ad.html」（a.htmlをバッチページ生成して生成されたテンプレート）とデータ（ITEM=I000a）から、テンプレートA（a.html）中のバッチページ生成後も命令遅延命令により残った命令にみ処理し、動的にユーザの要求するページを生成し、WWWサーバ手段11へ渡す（ステップ2024）。動的に生成されたページをWWWサーバ手段11がネットワーク2を通して、ページの要求もとのユーザのパーソナルコンピュータ3にHTTPプロトコルで送信する（ステップ2025）。送信されたページを、ユーザがページ要求したWWWブラウザ35により表示する（ステップ2026）。ページ表示が終了する（ステップ2027）。

【0050】

図15は、遅延命令付きページ生成手段23の動作を示す図である。TEMPLATE Aは、遅延命令付きページ生成手段23を、バッチページ生成制御手段13の制御のもと動作し、「\$NOP」命令がないアンダーラインの命令が実行され、データに展開され生成される。「\$NOP」命令は、この「\$NOP」がとれた新しい命令としてHTML中に展開される(TEMPLATE A')。図15の例のTEMPLATE A'中の\$ユーザ名がそれである。このバッチ生成されたTEMPLATE A'を、ユーザが動的にページ生成する要望をだすと、\$ユーザ名のみ呼出したユーザの名前にかわり生成され図15の生成ページとなる。この生成ページは、ユーザがTEMPLATE Aで呼出した場合と同じページであり、バッチページ生成されている文、TEMPLATE A'を呼出す場合のほうが、高速にページが生成され、高速に要求ページにアクセスできる効果をうむ。

【0051】

本実施例では、1つのテンプレート内部に、サーバ内部の状態であるデータ22に依存するページ生成の部分と、ユーザ指定の情報により生成される部分(例、アクセスしてきた、ユーザのユーザ名のページ合成等)が、共存している場合に有効である。すなわち、データ22に依存する部分のみ、事前にバッチページ生成して、第2のテンプレートにして、本来のユーザ要求時にユーザの情報により生成ページを完成させるページ生成方法が提供できる。これにより、共通部分の事前生成が可能となり、時間がかかる命令を削減することができ、個別ユーザ向けページの生成を高速化である。また、ユーザ名以外に、もっと一般的な用途としては、生成段階で確定する情報の展開を遅延させるために使用する方法である。例えば、現在の時間や、アクセス回数等のページ生成する瞬間のデータによりページ内容が作られる場合等、生成を遅延しておき、第2のテンプレートの実行時に、展開する事例が考えられる。

【0052】

上記実施例は、バッチページ生成制御13が、データ更新手段14のデータ更新時に毎回動作して、バッチページ生成していた実施例であったが、本発明は、

これに限定されるものでない。すなわち、バッチページ生成制御13は更新トリガを受け付けても内部に持つ更新フラグのみをセットして、実際のバッチ処理は、更新フラグがたってから、ユーザからのページ要求が初めてくるまで遅延させておく方法でもよい。更新フラグは、生成するページの分だけ存在し、生成ページが最新か、バッチ生成するべきかを識別するフラグとして使用する。以下、図16を使用して、その動作を詳細に説明する。図16のバッチページ生成制御手段13は、データ更新手段14がデータ22を更新するごとに、データ更新手段14から更新トリガを受け取る。または、タイマ手段16から更新トリガを受け取る。この際、データ22の個々と関連付けたデータの更新フラグ104を「セット」状態にする。セット状態でページ生成手段100から呼出しがかかると、対応するテンプレートA101をバッチ生成し、第2のテンプレートA'102を生成する。その際、バッチページ生成制御手段13の生成遅延機能103は、バッチ生成が正常に行われたことを確認して、対応する更新フラグを「リセット」とする。リセット状態で、ページ生成手段100からの呼出しがかかっても、なにもせずに返す。いずれにせよ、ページ生成手段100は、バッチページ生成制御手段13の生成遅延機能103を呼出すことで、TEMPLATE A'が最新であることこの保証を得られ、TEMPLATE A'を入力とした動的なページ生成をすることができる。

【0053】

実際の動作を図17のフローチャートを使用して詳細に説明する。

【0054】

ステップ3000からステップ3004までの処理は、ユーザ要求とは独立に、データ22の更新により開始される。データ更新手段14がDBMS15内のデータを更新する（ステップ3001）。データ更新手段14は、更新取りだをバッチページ生成制御手段13へ通知する。各バッチページ生成定義データ20に対応して存在する生成遅延機能103内部の更新フラグ104のうち、更新されたデータ関連する更新フラグ104をOFF状態にする（ステップ3003）。これで、ページのバッチ更新フラグ設定を終了する（ステップ3004）。次ぎに実際にユーザがWWWブラウザ35からページ要求した場合の処理（ステッ

3020からの処理)を説明する。ユーザのパーソナルコンピュータ3のWWWブラウザ35からページを要求する(ステップ3021)。バッチ生成されたテンプレート名称A'のテンプレート102による、ページ生成手段100を用いたページ生成要求を、WWWブラウザ35からWWWサーバ手段11へ出す(ステップ3022)。WWWサーバ手段11は、ページ生成手段100を起動する(ステップ3023)。ページ生成手段100は、バッチページ生成制御手段13の内部の生成遅延機能103に、前記指定されたテンプレートA'が最新のテンプレートが既に生成完了であるかを問合せる(ステップ3024)。更新フラグがONの場合は、生成が終了しており最新のテンプレートA'であるので、ページ生成手段100は、テンプレートA'102とデータ22から、ページ生成を行う。実際には、テンプレートA101の命令バッチ実行され、遅延命令により実行が遅延された命令がテンプレートA'の残存しており、この命令が実行されユーザが要求したページを動的に生成する(ステップ3026)。動的に生成されたページをWWWサーバ手段11がネットワーク2をとおしてユーザのパーソナルコンピュータ3に送信する(ステップ3027)。転送されたページをWWWブラウザ35が表示する。一方、ステップ3025の条件判定で、更新フラグ104がOFFの場合(データ更新があり、バッチ生成されたテンプレートがすでに、前回生成したテンプレートが古くなった場合)の処理を説明する。更新フラグがOFFであった場合、バッチページ生成制御手段13は、バッチページ生成定義データ20から生成するべきページを決定する(ステップ3030)。決定されたページを、テンプレートA101とデータ22を指定してページ生成手段100で生成する。その際、遅延命令があるテンプレート内命令は遅延命令がとれた命令として生成される(ステップ3031)。生成ページがテンプレートA'102として記憶する姪署と位置を、バッチページ生成定義データ20から決定する(ステップ3032)。生成したページをページ生成手段100が参照するテンプレートA'102としてディスク記憶装置7の記憶エリアに記憶する(ステップ3033)。上記のステップにより、テンプレートA'102が最新のテンプレートになったため、対応する更新フラグをONにする。このONにより、ステップ3025の条件判断がON状態と同等になるため、ステップ

3034の後、ステップ3026行こうを実行する。このように、ユーザ要求のページを高速に参照できる。

【0055】

本実施例の効果は、遅延バッチ生成により、バッチ生成の回数を削減できる効果がある。ページサーバ処理装置1のCPU負荷やファイルI/O付加を軽減できサーバの応答性能を劣化させない効果がある。

【0056】

上記実施例は、一般的なWWWサーバサイトにおける実施例であったが、本発明は、これに限定されるものでない。すなわち、WWWサーバを、エレクトリックコマースで使用する電子モールシステムへ適用することが可能である。すなわち、電子モールシステムにおける、入力される商品データから動的に商品カタログを生成したり、商品検索を行うシステムへ適用可能である。図18が、本ページ生成方法を、電子モールシステムへ適用した実施例である。上記の実施例のいずれも適用できるが、例えば、本稿の第1の実施例を適用した場合、商品データが少ない間は、すべての商品カタログをデータの更新とともにバッチ生成するように、バッチページ生成定義データ20に指定して、使用することができる。この場合、出店者クライアントPC201からデータ22の更新依頼がサーバに届くと、データ更新手段14が起動され、データ22を更新するとともに、バッチページ生成制御手段31を呼出す。バッチページ生成定義データ20によりすべての更新データに関連するページがバッチ生成され、WWWサーバ11から参照可能な記憶エリアに記憶される（テンプレート群30の記憶エリア）。実際の顧客は、ページ生成が事前に行われたページをアクセスして電子モールシステムの商品ページを高速に参照できる。また、検索は、検索の候補値を含むページが事前にバッチ生成され記憶されており、商品検索開始ページを高速にアクセスすることが可能である。検索を実行し結果のページを構築する部分は、従来の動的なページ生成でおこなう。また、商品データ量が多量の場合（例えば、数十万件以上）の場合、商品カタログを事前にすべてバッチ生成する方法では、サーバ記憶装置3が多量のディスク容量を必要とするため、すべてバッチ生成するのではなく、バッチページ生成定義を、頻繁に生成するページや1ページ生成に時間がかかる

るページを、バッチ生成するように指定して使用する。

【0057】

実際の動作を図18および図19のフローチャートを使用して詳細に説明する

【0058】

まず、ステップ4000からステップ4006のステップについて説明する。電子モールサーバ202内部の商品データ22が出店者PC201からの商品データ更新命令や電子モールシステム内部における商品データの更新により起動する（ステップ4000）。出店者PC201のWWWブラウザ20を使用して更新する商品データ22を電子モールサーバ202のコンピュータへアップロードする（ステップ4001）。WWWサーバ手段11が商品データ22を受け取り、データ更新手段14に渡す（ステップ4002）。データ更新手段14は、商品データ22の記憶されている商品データベースを更新し、更新完了の更新取りだをバッチバッチページ生成制御手段31に渡す（ステップ4003）。バッチページ生成制御手段31は、バッチ生成定義データ20に基づき、ページ生成手段23を用いて、テンプレート101と更新された商品データ22から商品ページを生成する（ステップ4004）。生成した商品ページをページ生成手段23が参照できるディスク記憶装置7上の位置に記憶する（ステップ4005）。これで、ページのバッチ生成を終了する（ステップ4006）。次ぎに、ユーザ商品ページを要求する処理について、ステップ4020から始まる処理で説明する。顧客の商品ページに要求により開始される（ステップ4020）。顧客のパソコンコンピュータ200のWWWブラウザ20から商品ページを要求する（ステップ4021）。WWWサーバ手段11は、ページ生成手段23を起動する（ステップ4022）。ページ生成手段23で、生成した生成商品ページ102をテンプレートとし、商品データ22とから動的に所品ページを生成する。ここでいう、生成商品ページ102は、ステップ4000から始まるバッチページ生成で生成されテンプレートとして記憶されたページである。動的に生成された商品ページをWWWサーバ手段11がネットワーク2を通してユーザのパソコンコンピュータ200に送信する（ステップ4024）。送信された商品ページをW

WWブラウザ2.0が表示する。これにより、ユーザ要求の商品ページが高速に参照できる。

【0059】

本実施例のように、本発明のページ生成方法を、電子モールシステムへ適用することで、商品データの入れ替えにフレキシブルに対応し、かつ、実際に顧客がアクセスする場合に、特に時間のかかるものや、よく使用するページが、事前にバッチ生成されているので、顧客は、商店者が更新した最新のデータに基づく最新の商品カタログページを、高速にアクセスでき、快適にショッピングを楽しめるという利点がある。

【0060】

上記の実施例は、WWWサーバでのページ生成方法に関するものであったが、本発明は、これに限定されるものではない。すなわち、ページの雛形であるテンプレートの集合と、該ページを構成するコンテンツであるデータの集合と、該テンプレートを入力とし、該テンプレート中に記述されている該データ集合へのデータ取得命令を起動し、がデータを取得し、該テンプレート記述と該取得データからページを生成するページ生成手段と、生成するページに関する情報を指定したバッチページ生成定義と、該バッチページ生成定義を入力として、該ページ生成手段を制御し、バッチ型のページ生成を行うバッチページ生成制御手段とを、例えばCD-ROM等のページ生成用記憶媒体に記憶し流通させ、ページアクセスユーザがページをアクセスする前に、該バッチページ生成制御手段を起動して、アクセスするページを生成させ、生成したページを参照させるページ生成方法が可能である。

【0061】

本実施例では、重複するページの雛形部とデータ部が独立しており、コンパクトに過般媒体（例、CD-ROM等）に記憶して、流通させることが可能である特長がある。また、インストールの段階や、実際にページアクセスが必要になった段階で、バッチ型のページ生成を起動させ、ページを生成して必要なページを作り出し記憶されるため、実際の参照段階では、高速にページにアクセスできる効果がある。また、データ部のみ新規文分や差分が、ネットワーク経由や、新た

な流通可能な記憶媒体で得られれば、また、新しいページをバッチ生成してページを追加する能力も上記ページ生成用記憶媒体は機能としてもつ。

【0062】

上記の実施例の全体を通して、HTMLで記述されたページに関してのべてきたが、本発明は、これに限定されるものではない。すなわち、キャラクタの並びであるテキストデータで表現できる形式（XML、ポストスクリプト、SGML等）のもの全ては、文字列置換であるページ生成方法が適用可能であり、テキストベースのあらゆる形式のページは、テンプレートと動的なページ生成機能で生成されるページであるならば、上記実施例を適用することで高速化することが可能である。

【0063】

本発明のページ生成方法を、エレクトリックコマースシステムの電子モールサーバにおける動的なページ生成に適用し、その効果を測定した。いずれも、1ページ生成に多数のDB検索があり、取得したデータの展開が多く、生成時間を見るものである。

【0064】

サンプル1：商店一覧ページ（データ呼び出し：15回）

サンプル2：絞り込み検索ページ（データ呼び出し：8回）

サンプル3：商品一覧ページ（25回）

実測データ：（表示完了までの時間：10人が同時にアクセスした場合のレスポンス）

対象	図22の場合	図1の場合	効果
サンプル1：商店一覧ページ	： 15秒	3秒	5倍
サンプル2：絞り込み検索ページ	： 40秒	4秒	10倍
サンプル3：商品一覧ページ	： 45秒	3秒	15倍

本方法により、ページ生成の時間が短縮でき、ページアクセスの全体のレスポンスを、2.5倍～15倍向上できた。本方法は、ページ要求の前にページ内容のライフ時間に応じたページ生成がされているため、上記の測定結果のごとく、フレキシブルに変化するページに関して、従来、毎回動的に生成されるページ

で時間がかかったものも、実際の参照時に高速にページ生成されたページが得られ、高速なページアクセスが可能となる効果がある。また、ページが固定化できるため、従来のページアクセス技術（公知例1、2、4）が使用できるため、ページをクライアントまで転送するコストの削減もおこなえ、トータルなページアクセスの高速化を実現する効果がある。

【0065】

【発明の効果】

所定の更新契機に従いページを生成しておくことにより、送信要求を受け付けてから短時間で応答することが可能となる。

【図面の簡単な説明】

【図1】

本発明のページ生成方法のシステム構成図。

【図2】

本発明のページ生成方法が実行される計算機システム。

【図3】

本発明のページ生成方法の処理を示すフローチャート。

【図4】

ページのテンプレートの例。

【図5】

生成された後のページの例。

【図6】

生成された後のページのWWWブラウザの表示例。

【図7】

ページを構成するデータの記憶例。

【図8】

生成されたページのWWWブラウザの表示例。

【図9】

生成されたページのWWWブラウザの表示例。

【図10】

テンプレートと生成H T M L ページの例。

【図11】

バッチページ生成定義の例。

【図12】

本発明のページ生成方法の命令遅延型の実施例のシステム構成図。

【図13】

本発明のページ生成方法の命令遅延型の実施例が実行される計算機システム。

【図14】

本発明のページ生成方法の命令遅延型の実施例の処理を示すフローチャート。

【図15】

本発明のページ生成方法の命令遅延型の実施例のテンプレートと生成テンプレートと生成ページの例。

【図16】

本発明のページ生成方法の更新フラグ型の実施例のシステム構成図。

【図17】

本発明のページ生成方法の更新フラグ型の実施例の処理を示すフローチャート

【図18】

本発明のページ生成方法を電子メールに適用した実施例のシステム構成図。

【図19】

本発明のページ生成方法を電子メールに適用した実施例の処理を示すフローチャート。

【図20】

ページアクセス方法の公知例 (W W W システム)。

【図21】

ページアクセス方法の他の公知例 (プロキシ方式)。

【図22】

ページアクセス方法の他の公知例 (ページ生成方式)。

【図23】

ページアクセス方法の他の公知例（プッシュ方式）。

【図24】

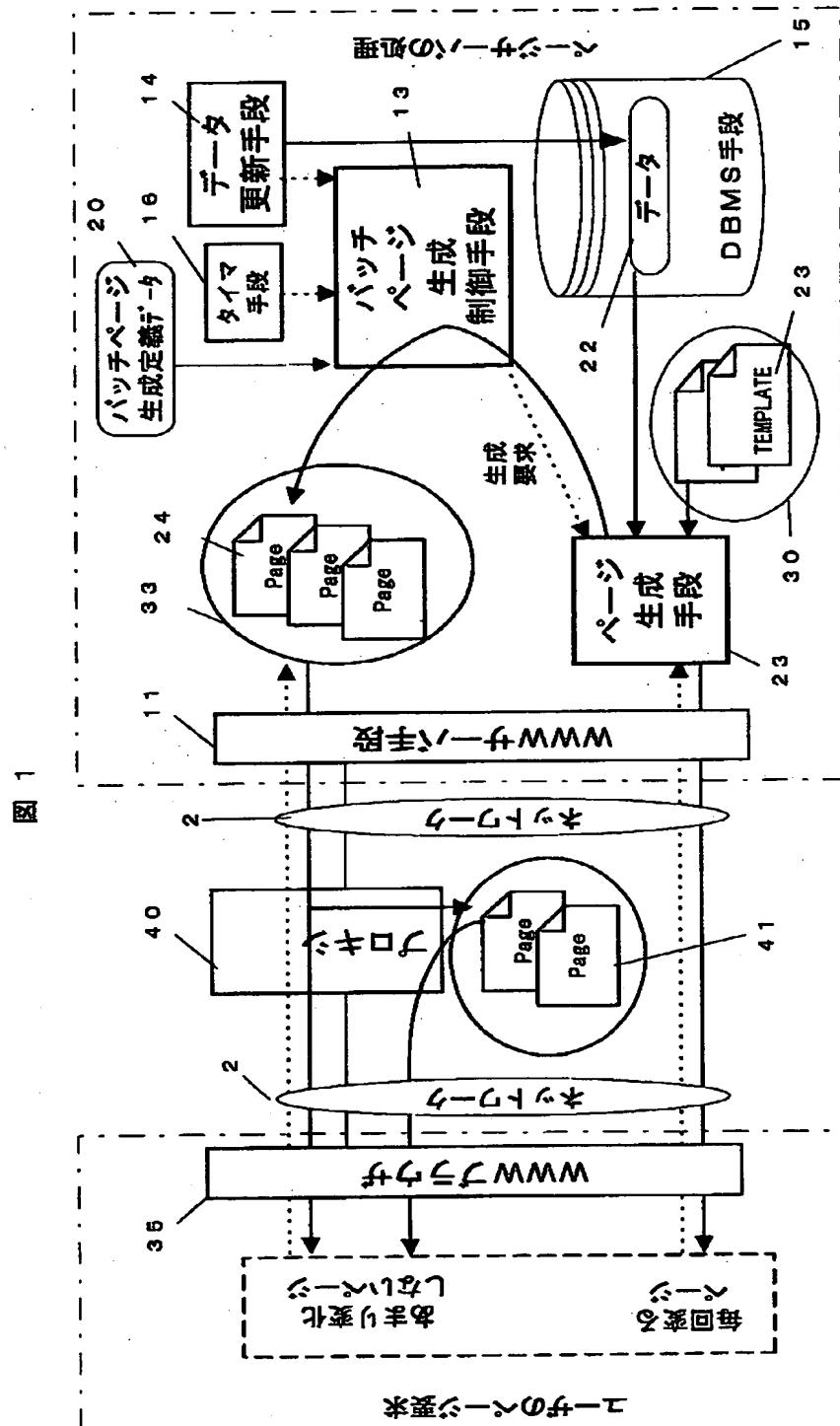
ページアクセス方法の他の公知例（予測取得方式）。

【符号の説明】

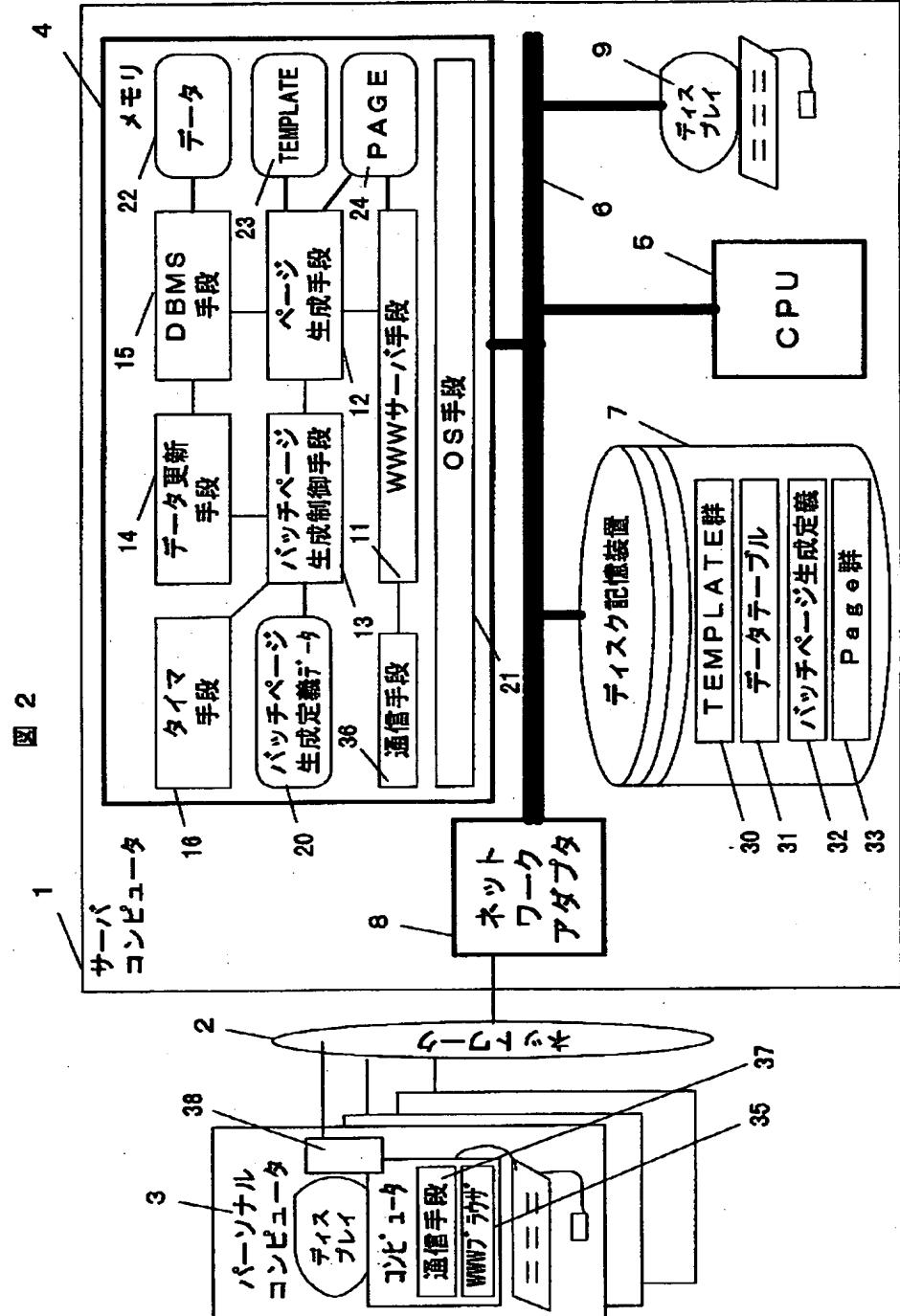
- 1 …サーバコンピュータ、
- 2 …ネットワーク、
- 3 …パーソナルコンピュータ、
- 4 …メモリ、
- 5 …C P U、
- 6 …バス、
- 7 …ディスク記憶装置、
- 8 …ネットワークアダプタ、
- 9 …ディスプレイ、
- 1 1 …WWWサーバ手段、
- 1 2 …ページ生成手段、
- 1 3 …バッチページ生成制御手段、
- 1 4 …データ更新手段、
- 1 5 …D B M S 手段、
- 1 6 …タイマ手段。

【書類名】 図面

【図1】

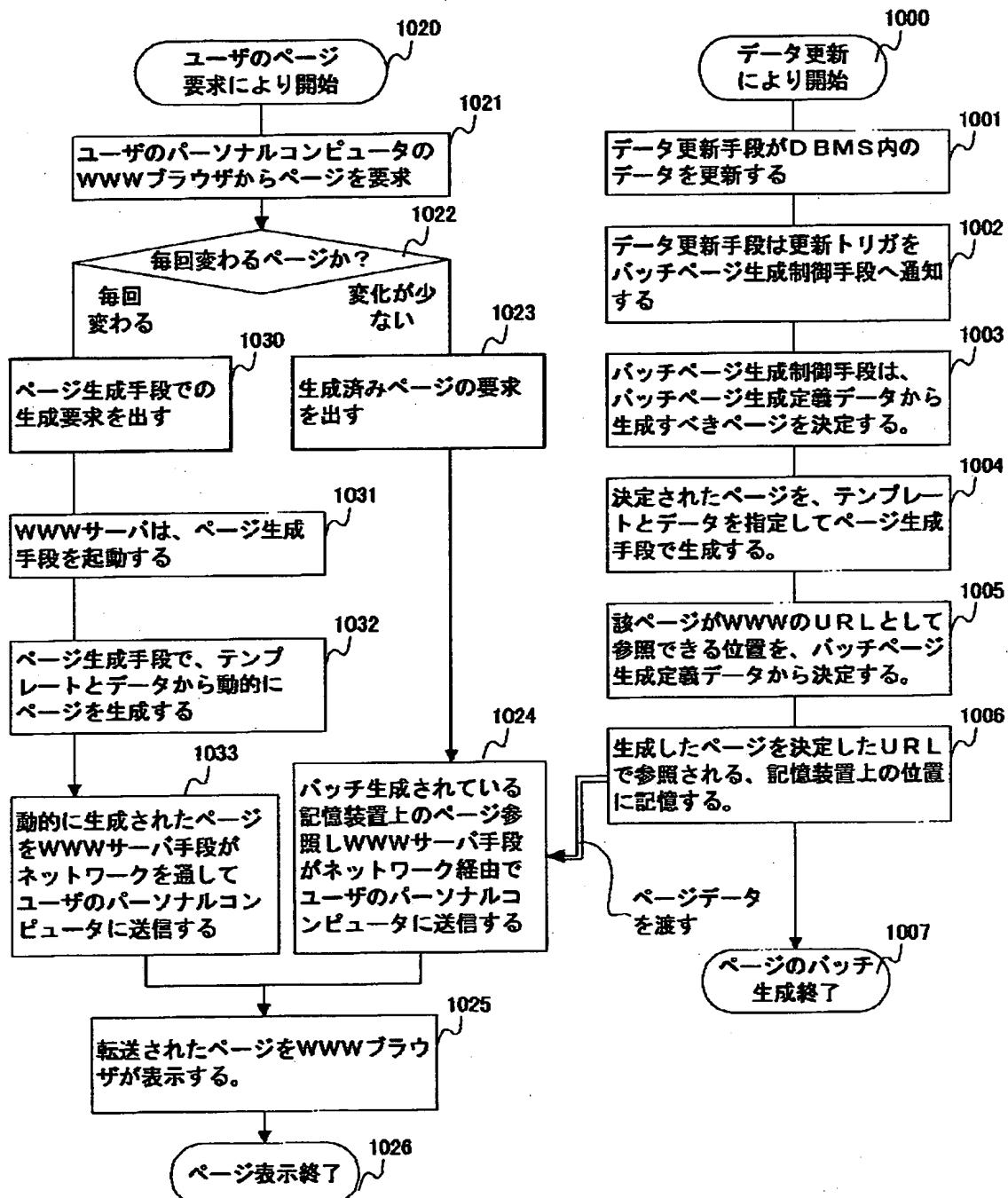


【図2】



【図3】

図 3



【図4】

図 4

```

<HTML>
<head> <title>商品検索ページ</title> </head>
<BODY>

<!>EXEC INTERFACE=PDBManager METHOD=GetAnchor
PARAM=("$ストアID|$商品ID|$商品名|$メーカー名|$販売価格|$メーカー型番|$標準価格|$キーワード",
       "$CONDITION", "$KEYWORD", 0, "", 0, 0, 1, 3)
RECORD=REC>

<CENTER>
<B><FONT COLOR="#3333FF"><H1>商品検索</H1></FONT></B>.
<B>ただいま検索した商品の件数は $COUNT </b><br>
<table border="5" cellpadding="3" cellspacing="0">
  <tr>
    <td width=20%>
      <H4><b>選択対象名</b></H4>
    </td>
    <td width=80% align="center">
      <H4><b>選択候補</b></H4>
    </td>
  </tr><tr>
    <td width=20%><b>大分類</b></td>
    <td><!>EXEC INTERFACE=PDBManager METHOD=GetAnchor
      PARAM=("$大分類", "$CONDITION", "$KEYWORD", 1, "大分類", 0, 0, 1, 1)
      RECORD=REC>
      <!>REPEAT RECORD=REC>
        <A href="/cgi-bin/page?SHOPID=$SHOPID&TERTPLATE=searchmain.html"
           &KEYWORD=<#ENCODE ($KEYWORD)>&CONDITION=<#ENCODE ($NCOND)>>
           $VALUE</A>
      <!>REPEAT>
    </td>
  </tr><tr>
    <td width=20%><b>小分類</b></td>
    <td><!>EXEC INTERFACE=PDBManager METHOD=GetAnchor
      PARAM=("$小分類", "$CONDITION", "$KEYWORD", 1, "小分類", 0, 0, 1, 1)
      RECORD=REC>
      <!>REPEAT RECORD=REC>
        <A href="/cgi-bin/page?SHOPID=$SHOPID&TERTPLATE=searchmain.html"
           &KEYWORD=<#ENCODE ($KEYWORD)>&CONDITION=<#ENCODE ($NCOND)>>
           $VALUE</A>
      <!>REPEAT>
    </td>
  </tr> <tr>
    <td><H4><b>販売価格(円)</b></H4> </td>
    <td><!>EXEC INTERFACE=PDBManager METHOD=GetAnchor
      PARAM=("$販売価格", "$CONDITION", "$KEYWORD", 2, "販売価格", 0, 0, 1, 2)
      RECORD=REC>
      <!>REPEAT RECORD=REC>
        <A href="/cgi-bin/page?SHOPID=$SHOPID&TERTPLATE=searchmain.html"
           &KEYWORD=<#ENCODE ($KEYWORD)>&CONDITION=<#ENCODE ($NCOND)>>
           $VALUE1 ~ $VALUE2</A>
      <!>REPEAT>
    </td>
  </tr>
</table>
</center>
</body>
</html>

```

【図5】

図 5

```

<HTML>
<head> <title>商品検索ページ</title> </head>
<BODY>
<CENTER>
<B><FONT COLOR="#3333FF"><H1>商品検索</H1></FONT></B>
<b>ただいま検索した商品の件数は 184 </b><br>
<table border="5" cellpadding="3" cellspacing="0">
  <tr>    <td width=20%>
    <h4><b>選択対象名</b></h4>
    </td>
    <td width=80% align="center">
    <h4><b>選択候補</b></h4>
    </td>
  </tr> <tr>
    <td>
      <h4><b>大分類</b></h4>
      </td>
      <td>
        <A href="/cgi-bin/page?SHOPID=a-shop&TEMPLATE=mcsearchmain.html"
          &KEYWORD=&CONDITION=%91%E5%95%AA%97%DE+
          =+%27PDF%95%86%8F%91%27">
          文書 </A>
        <A href="/cgi-bin/page?SHOPID=a-shop&TEMPLATE=mcsearchmain.html"
          &KEYWORD=&CONDITION=%91%E5%95%AA%97%DE+
          =+%27%83%8A%83y%83C%83h%83J%81%5B%83h%27">
          音声 </A>
        <A href="/cgi-bin/page?SHOPID=a-shop&TEMPLATE=mcsearchmain.html"
          &KEYWORD=&CONDITION=%91%E5%95%AA%97%DE+
          =+%27%89%E6%91%9C%27">
          画像 </A>
        <A href="/cgi-bin/page?SHOPID=a-shop&TEMPLATE=mcsearchmain.html"
          &KEYWORD=&CONDITION=%91%E5%95%AA%97%DE+
          =+%27%93%AE%89%E6%27">
          動画 </A>
      </td>
    </tr>
    ...

```

【図6】

図6

商品検索

ただいま検索した商品の件数は 184

選択候補	
選択対象名	
50 大分類	文書画像動画
51 小分類	リソース クラシック サウンドトラック バスル ポピュラー ヨーロッパ ロック ハード ロック 音楽 紙自動車
52 販売価格(円)	0 ~ 49 50 ~ 99 100 ~ 299 300 ~ 499 500 ~ 4999 5000 ~ 9999

【図 7】

商品データテーブル

図 7

60

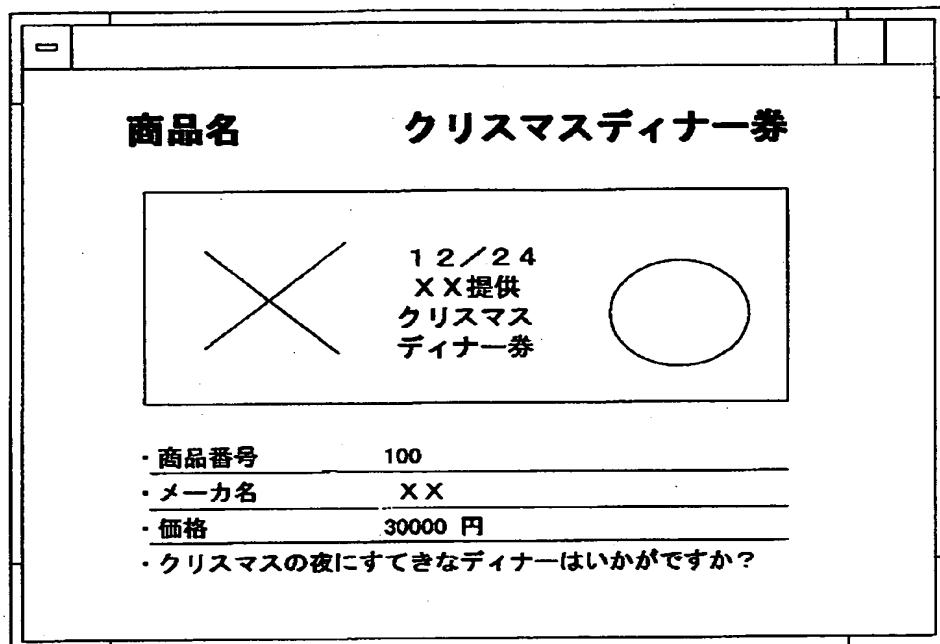
商品 ID	商店 ID	商品名	メーカー名	価格	会員価格	画像	説明
...							
1000a	AA社	クリスマスディナー券	XX	30000	25000	PP.gif	クリスマスの
1000b	BB社	クリスマスケーキ	YY	4000	3800	P1.gif	あまいケーキ
...			...				
...							
...							

61

62

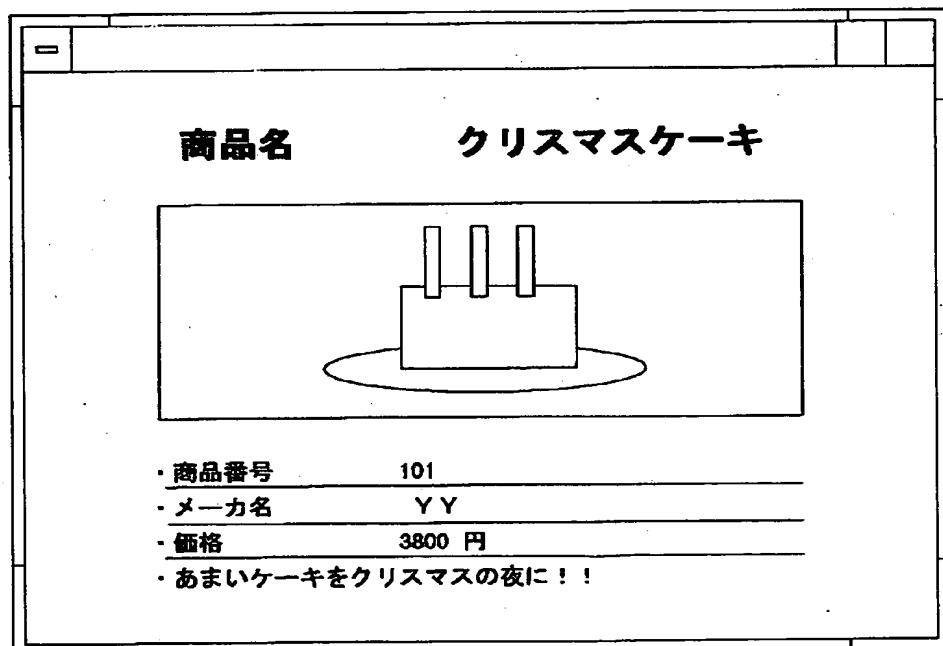
【図8】

図 8

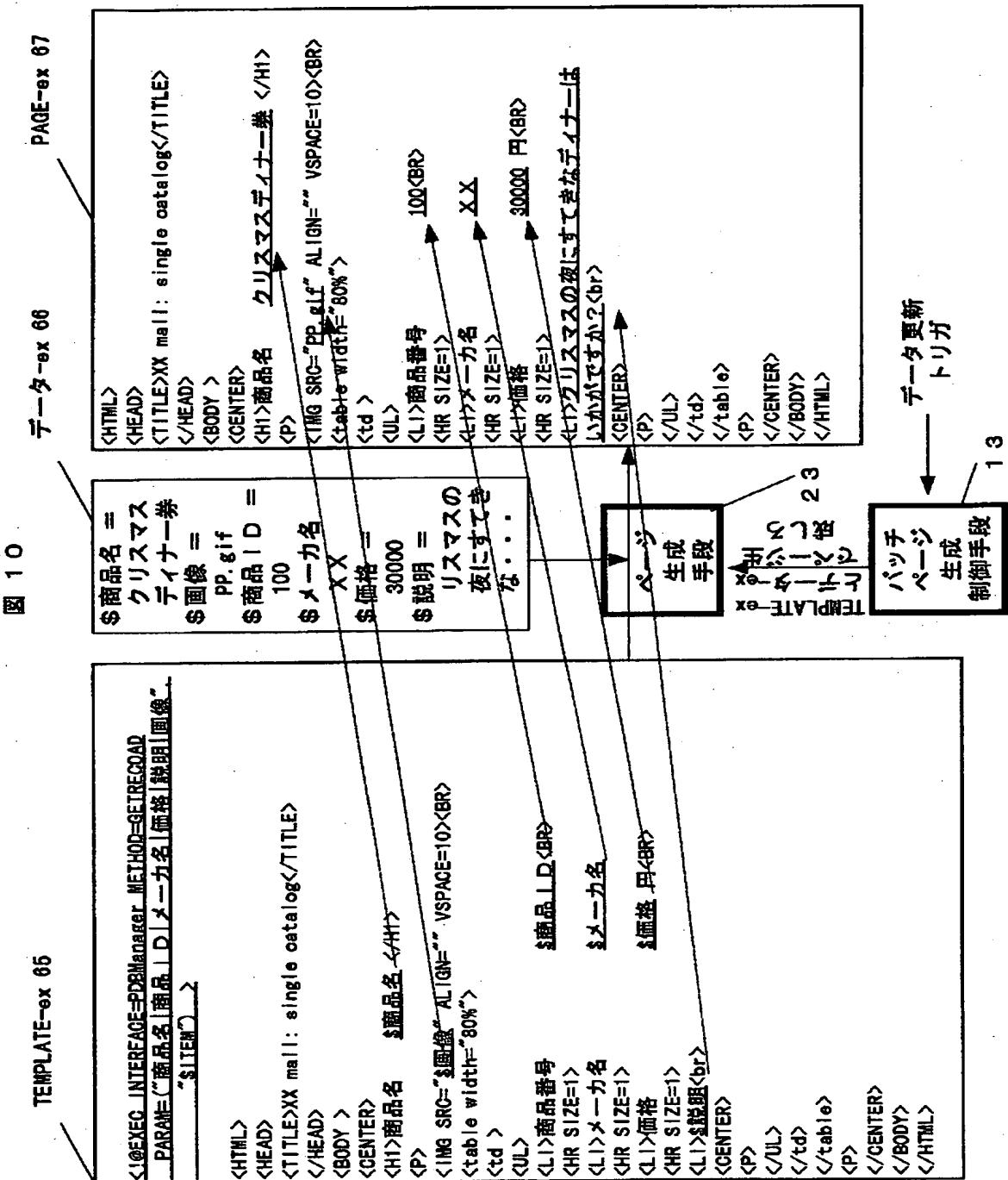


【図9】

図 9



【図10】



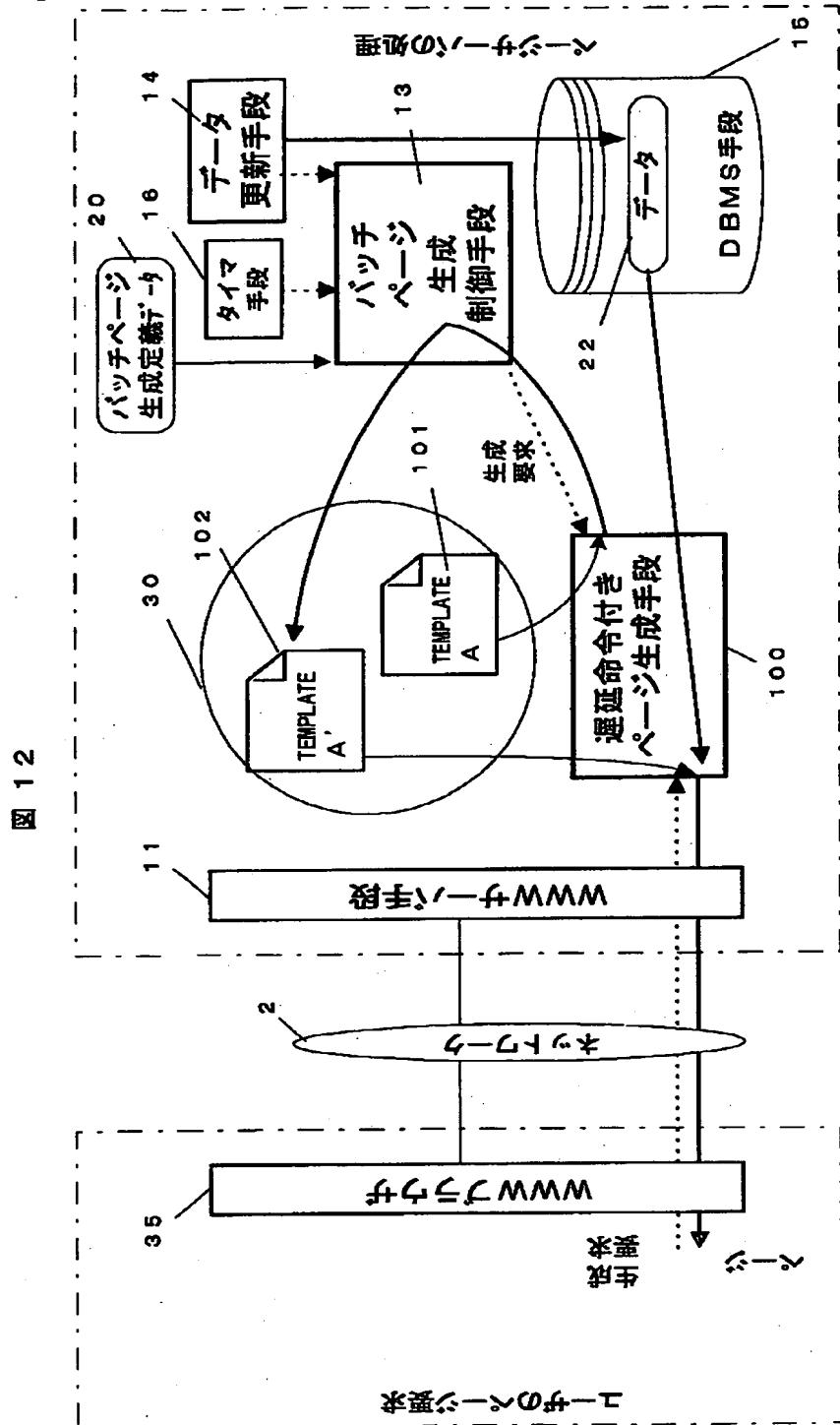
【図11】

図11

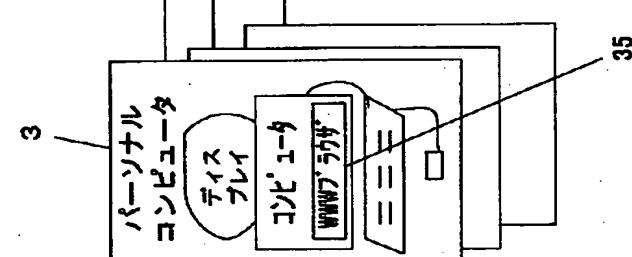
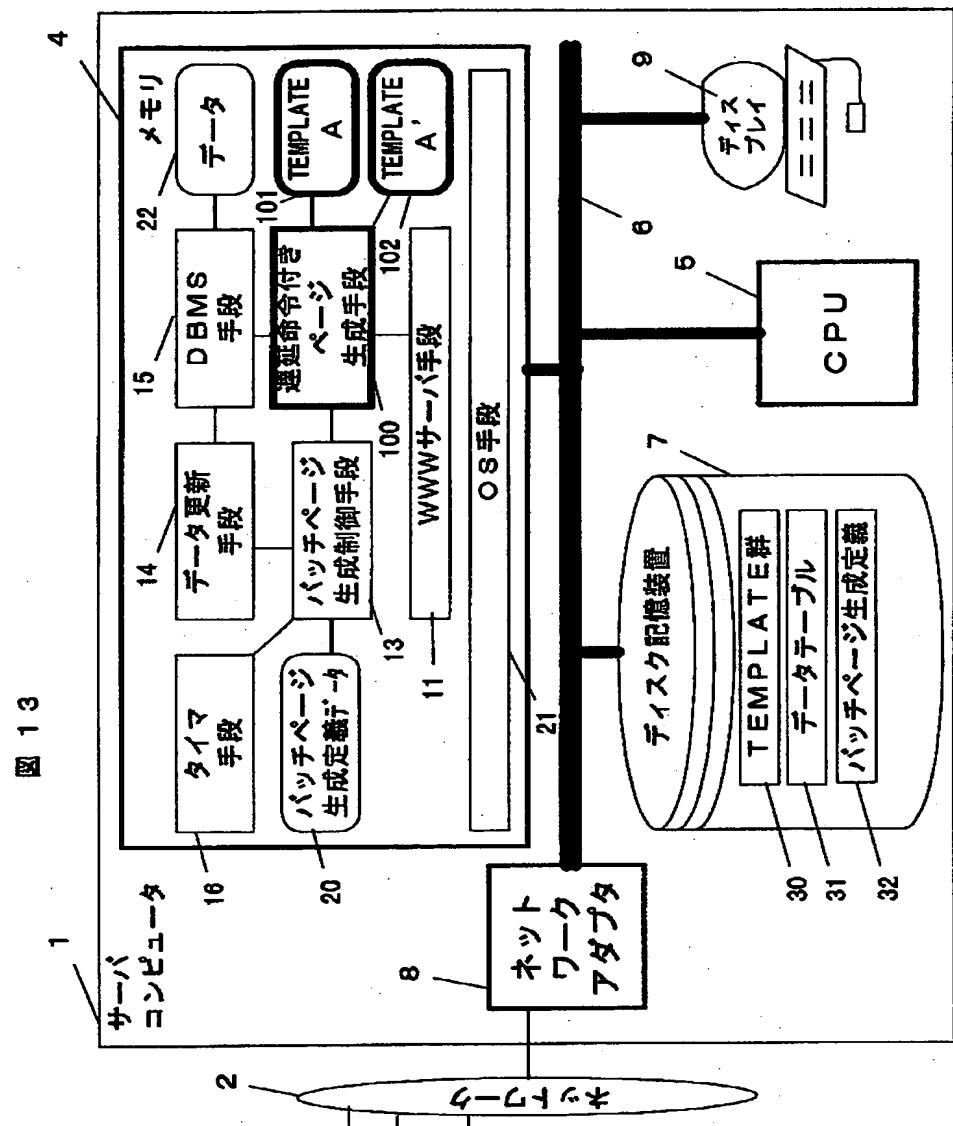
バッチページ生成定義 70

定義番号	トリガの種類	トリガの成立条件	バッチページ生成命令	生成ページ名
7_1	時間	毎朝 6 時	page TEMPLATE=c.html	c_g_.html
7_2	データ更新	A 商品データ更新	page TEMPLATE=a.html&ITEM=100a	a_g_100a.html
7_3	TEMPLATE更新	a.html 更新	page TEMPLATE=a.html&ITEM=*	a_g_*.html
4)		...		
		...		

【図1.2】

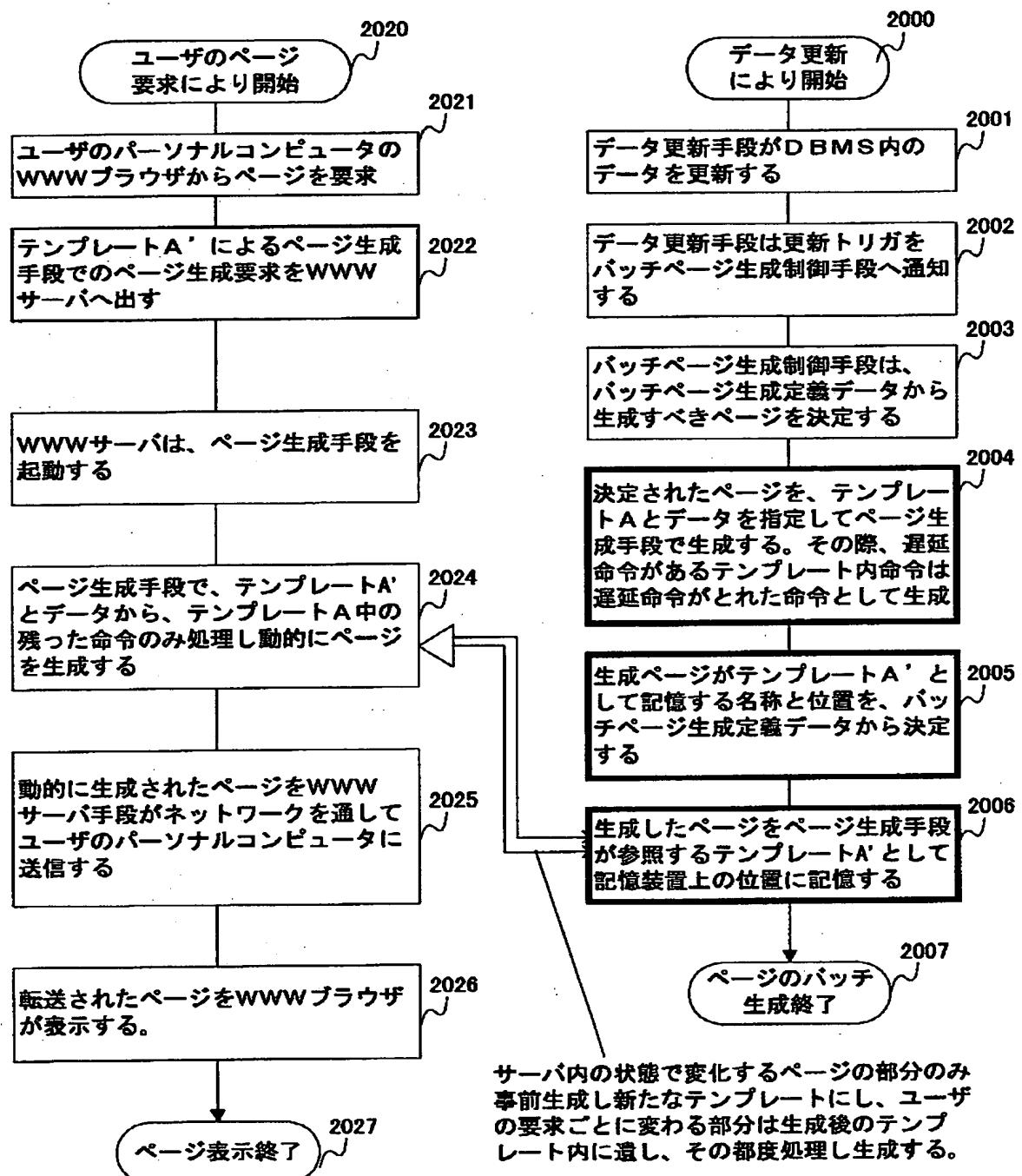


【図13】



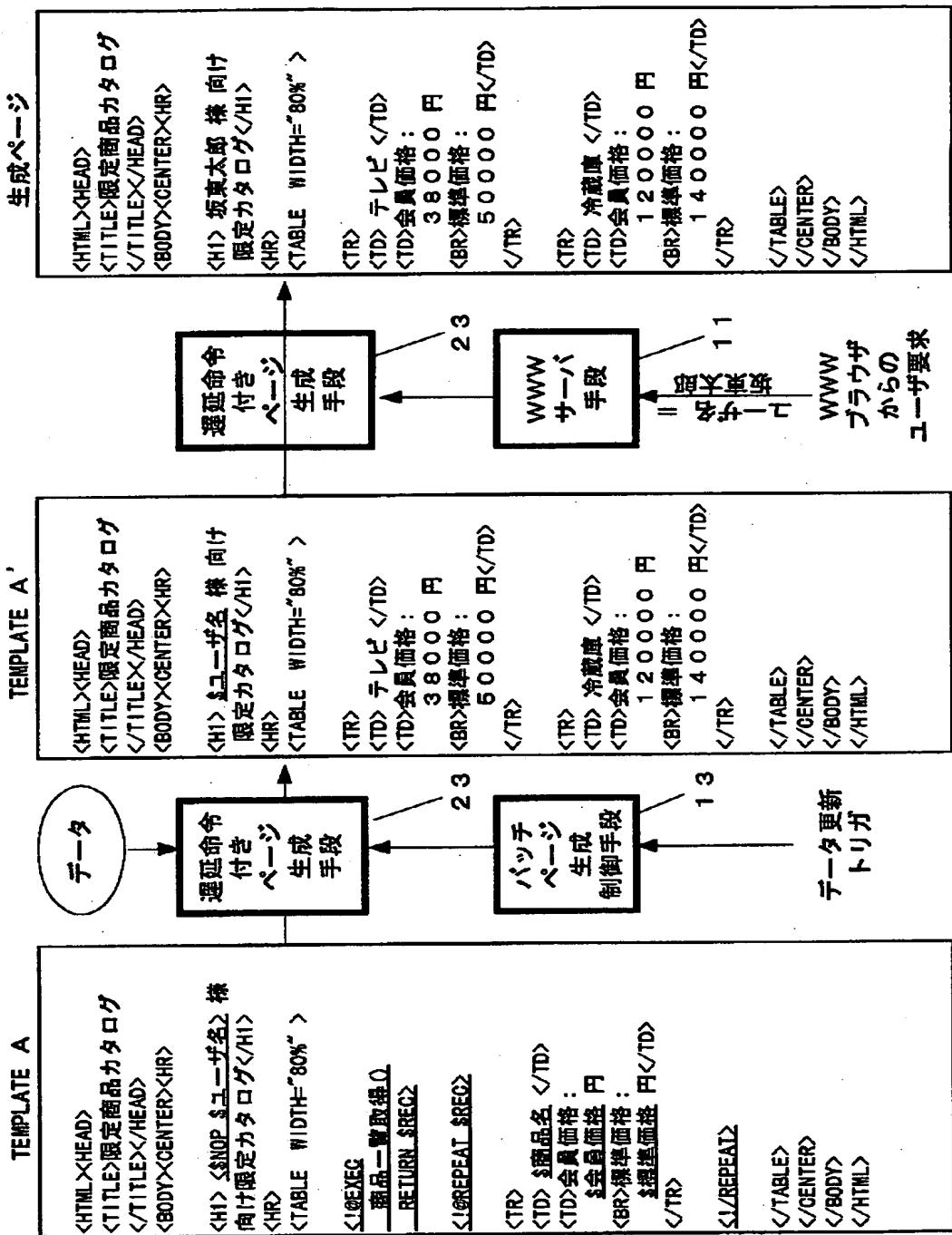
【図14】

図14

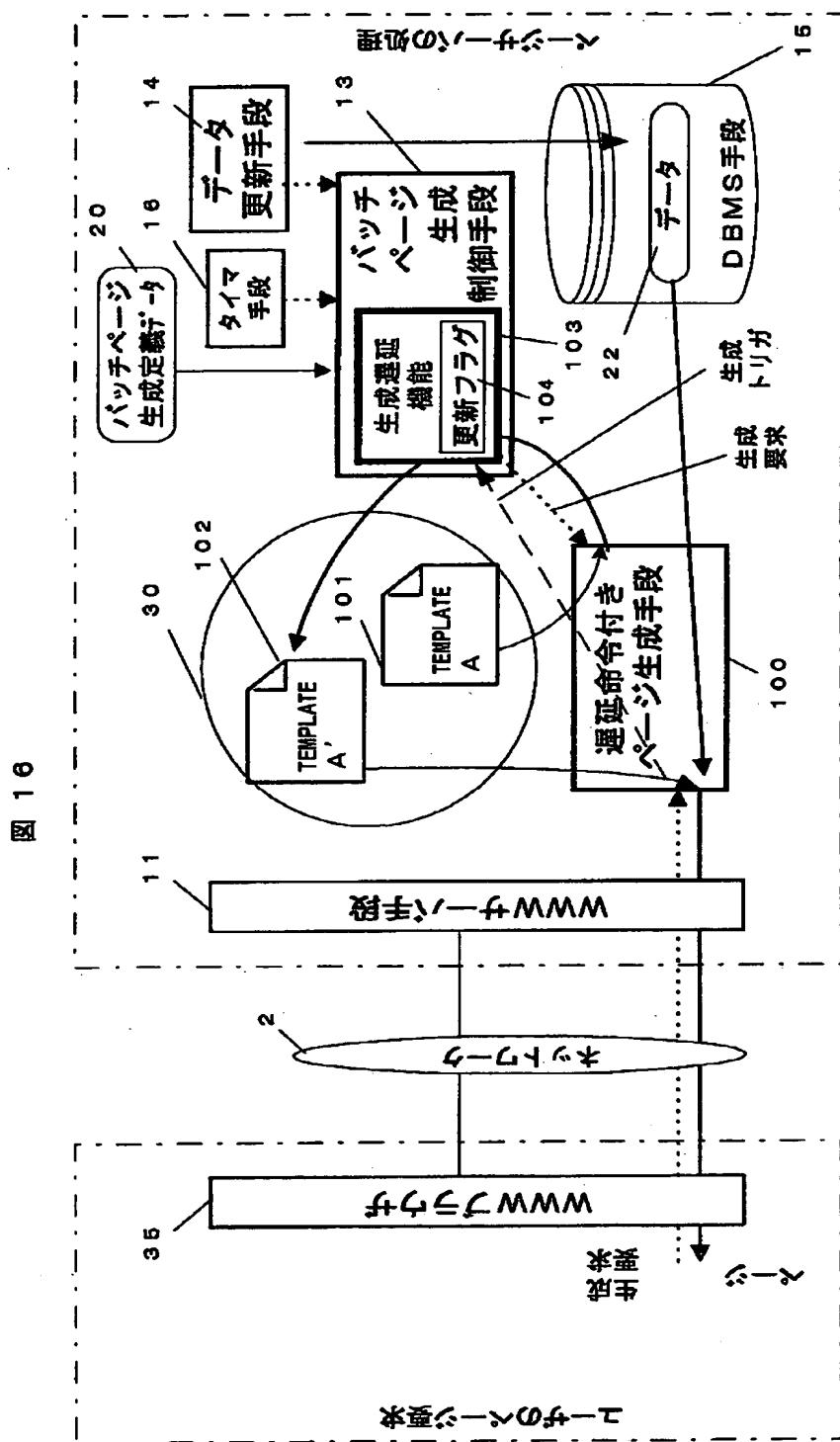


【図 15】

図 15

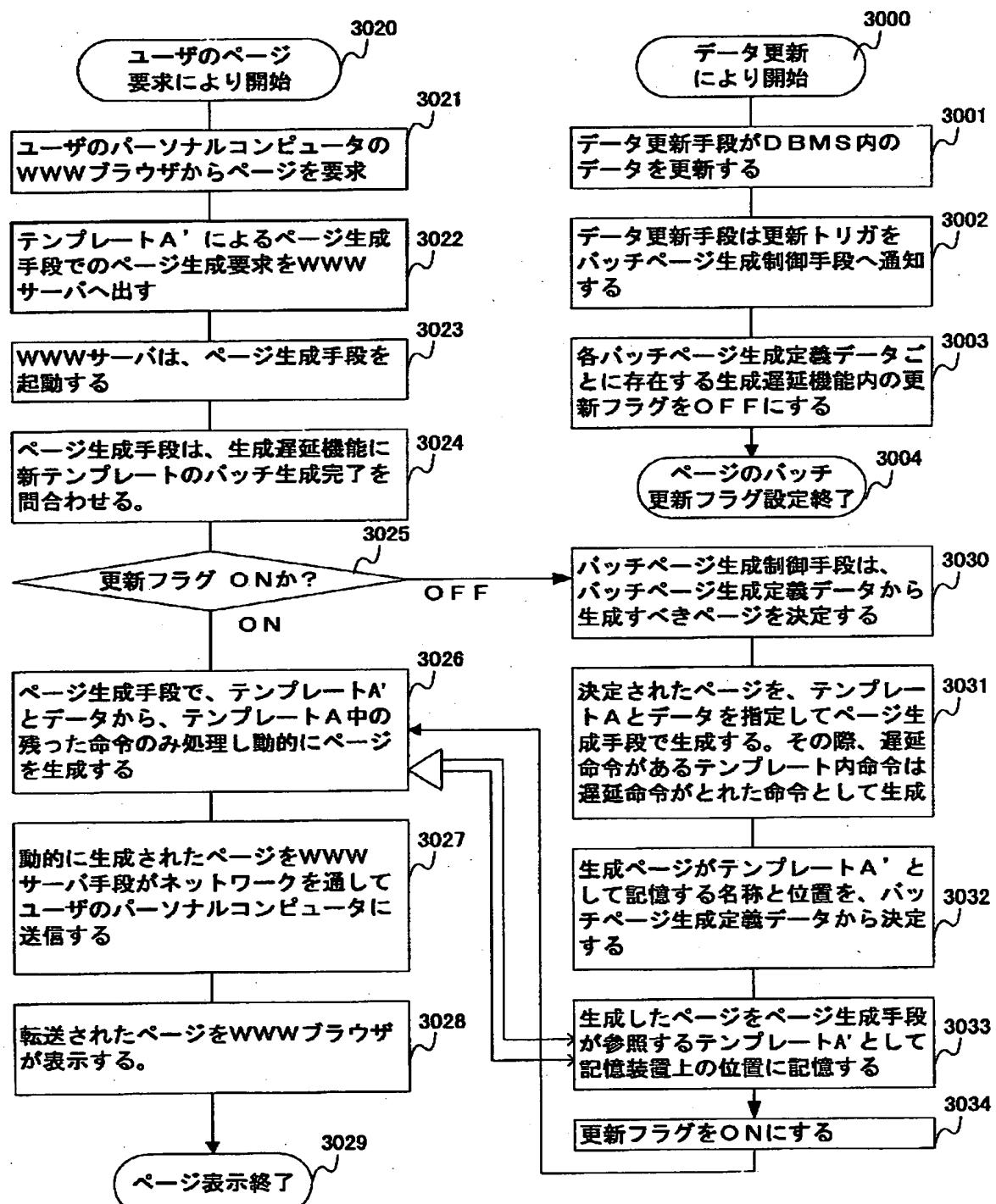


【図16】

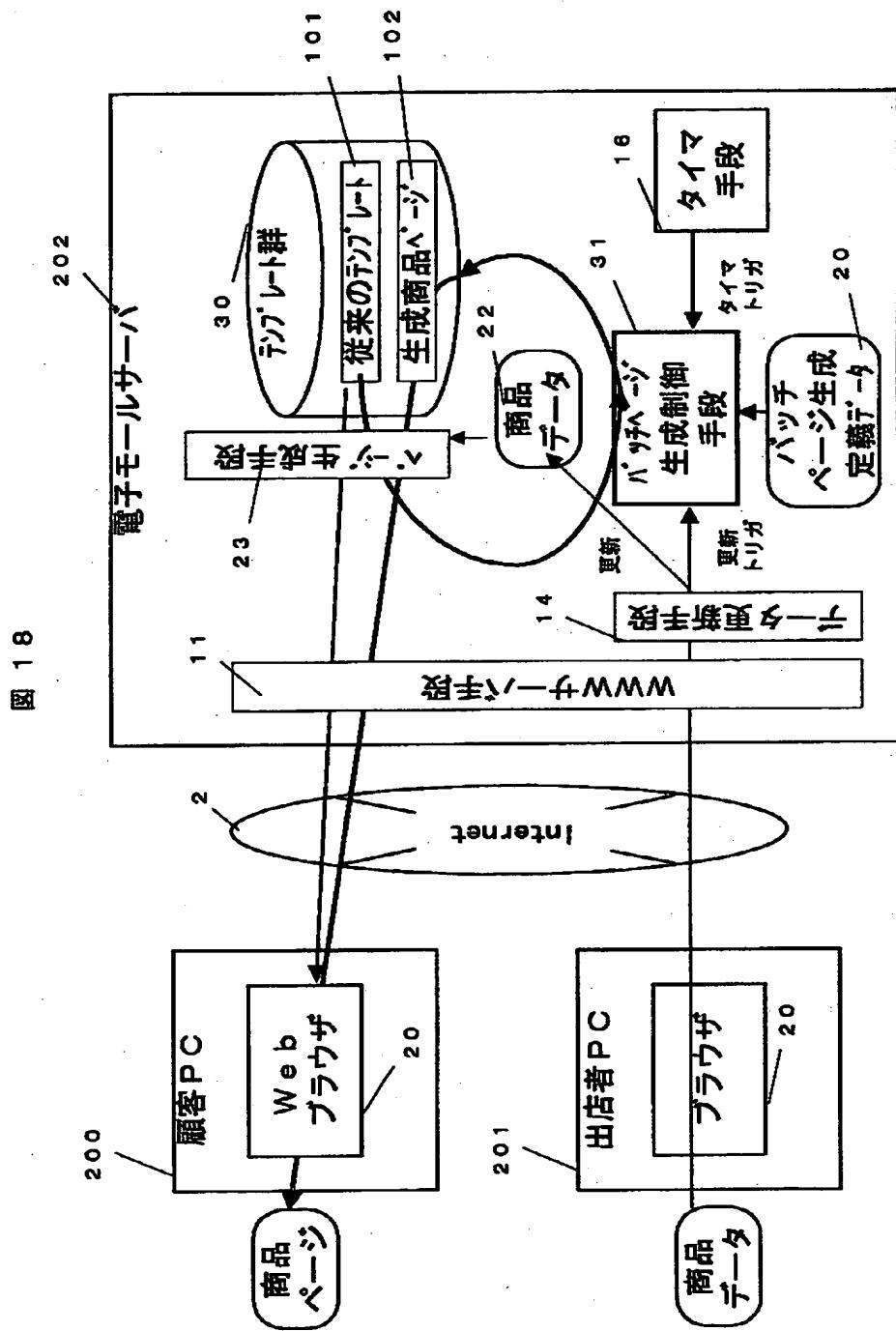


【図17】

図17

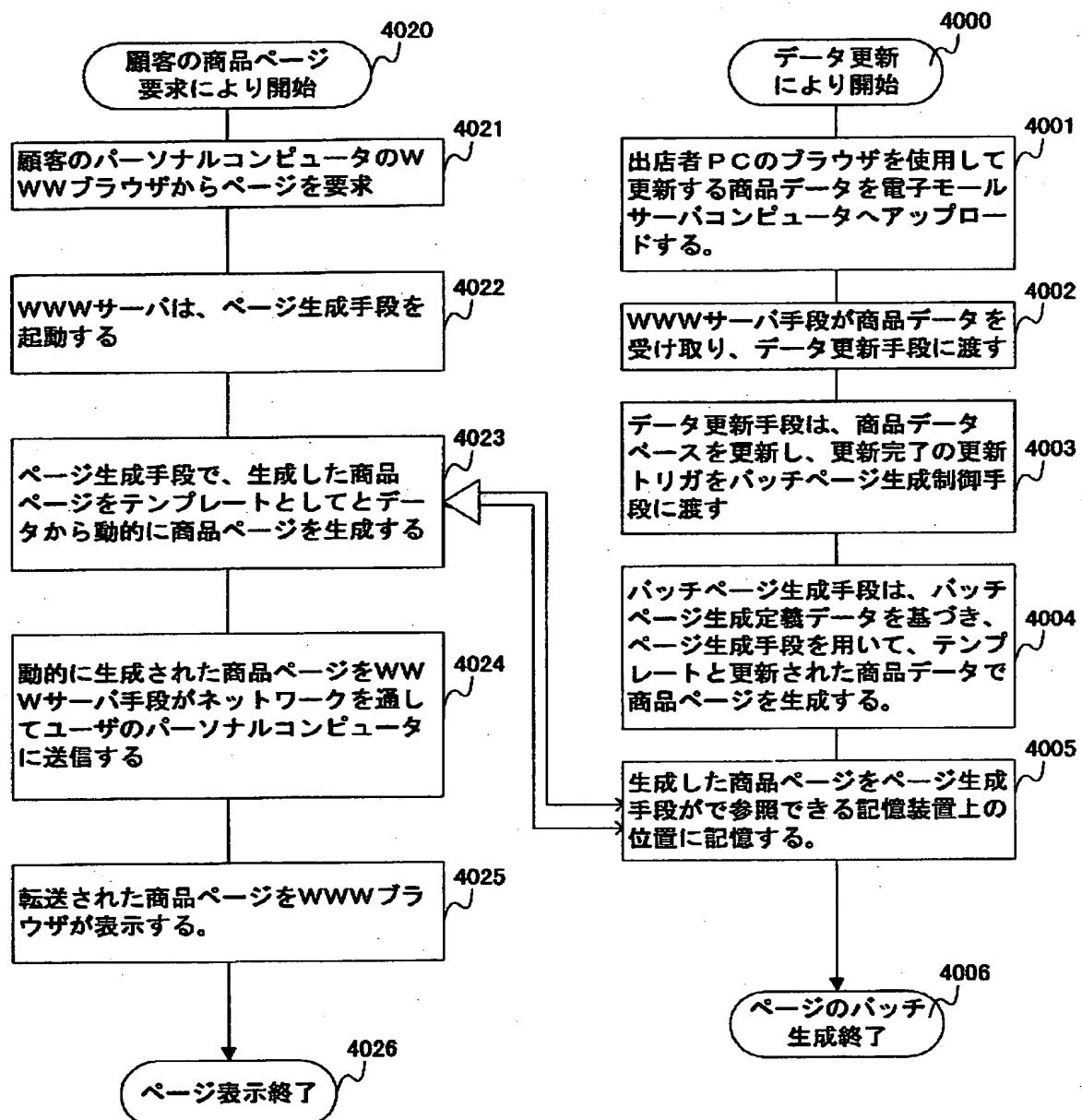


【図18】



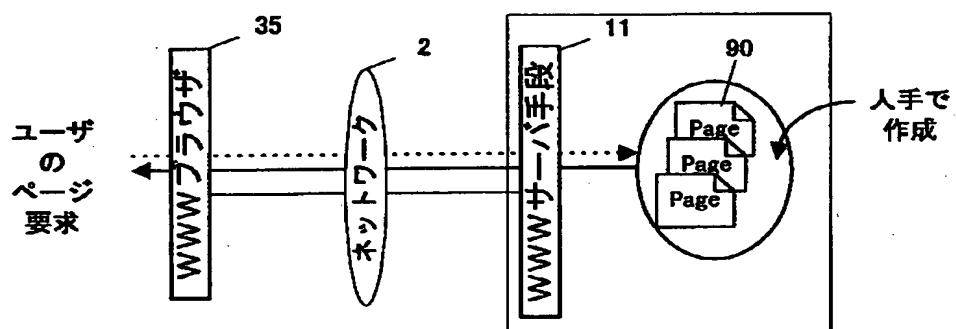
【図19】

図19



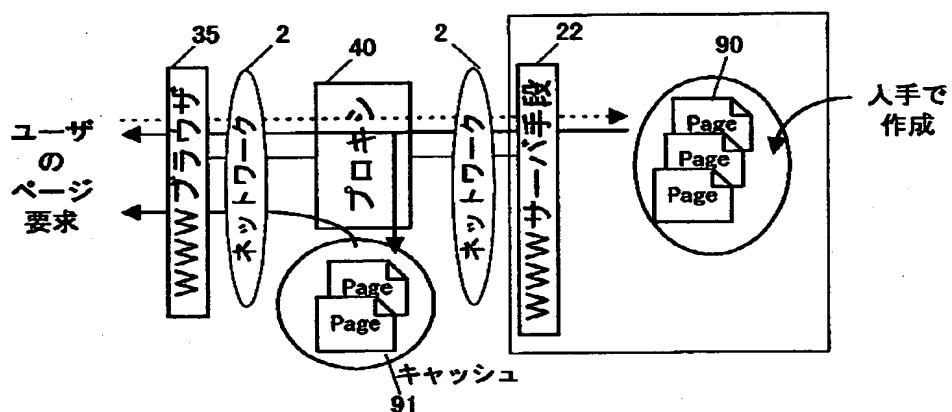
【図20】

図20



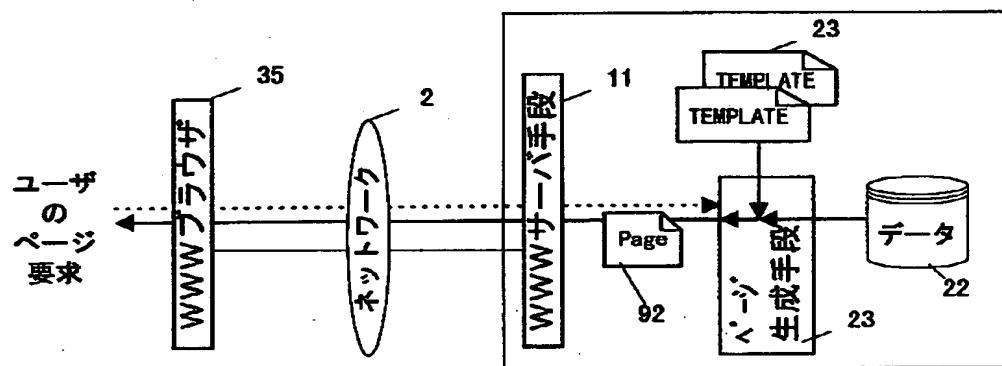
【図21】

図21



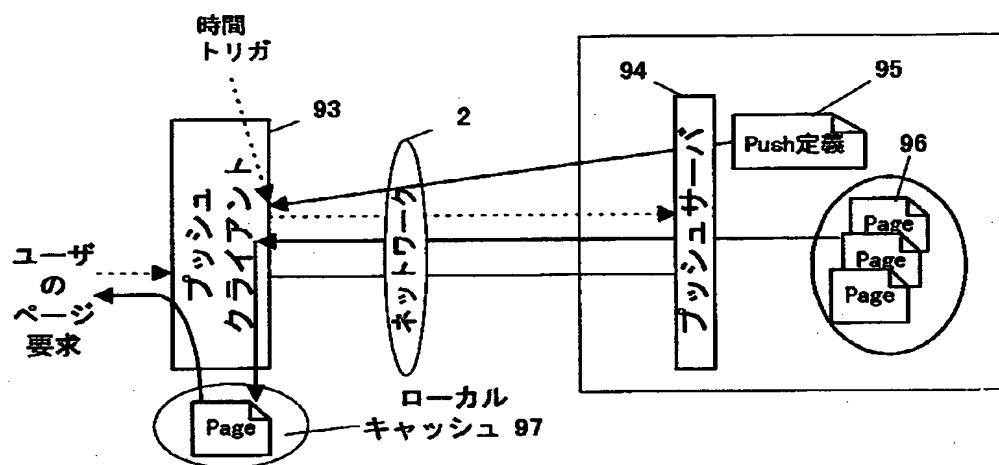
【図22】

図22



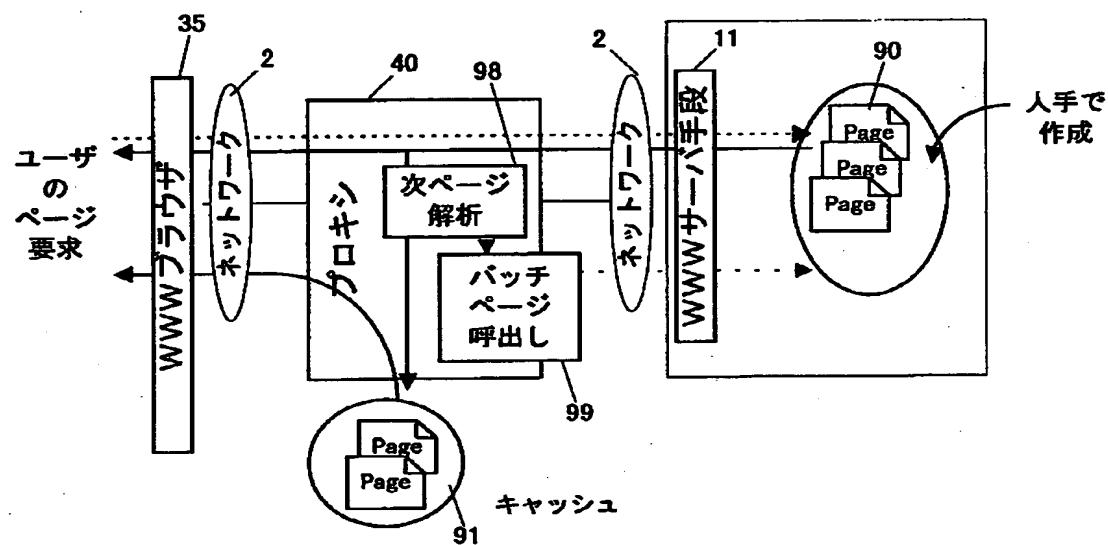
【図23】

図23



【図24】

図24



【書類名】 要約書

【要約】

【課題】

従来のWWWシステムでは、プロキシやプッシュ技術により高速なページアクセスが可能である反面、ユーザがページを全て作成する必要があり、また、固定的なページしか生成できない。一方、動的なページ生成方式では、テンプレートと埋め込むデータから各種ページを動的に生成するため、フレキシブルなページ生成ができる。このため、動的なページ生成方式が近年広い範囲で使用されるようになった。しかし、毎回、テンプレートの解析、データのデータベースからの検索&取得、ページの合成を行うため、ページアクセス時間が遅いという課題がある。

【解決手段】

ページ要素データの更新トリガに従い該ページ生成手段を起動しページを生成し、該ページサーバが参照可能な位置に該バッチ生成ページを配置するバッチページ生成制御手段を具備し、該ページサーバがページ要求を受け付けた時に、該バッチ生成ページをクライアントに送信する。

【選択図】 図1

出願人履歴情報

識別番号 [000005108]

1. 変更年月日 1990年 8月31日

[変更理由] 新規登録

住 所 東京都千代田区神田駿河台4丁目6番地

氏 名 株式会社日立製作所